Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations 1. Thesis and Dissertation Collection, all items

1971-12

A computer program for solving transient heat conducting problems.

Erbayram, Coskun.

Monterey, California ; Naval Postgraduate School

http://hdl.handle.net/10945/15618

Copyright is reserved by the copyright owner

Downloaded from NPS Archive: Calhoun

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

http://www.nps.edu/library

A COMPUTER PROGRAM FOR SOLVING TRANSIENT HEAT CONDUCTION PROBLEMS

Coskun Erbayram

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A COMPUTER PROGRAM FOR SOLVING TRANSIENT HEAT CONDUCTION PROBLEMS

by

Coskun Erbayram

Thesis Advisor: T. E. Cooper

December 1971

Approved for public release; distribution unlimited.

A Computer Program for Solving
Transient Heat Conduction Problems

by

Coşkun Erbayram
Lieutenant (junior grade), Turkish Navy
B.S., Turkish Naval Academy, 1966

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1971

Thesis
E5656
c.1

ABSTRACT

A computer program, with the code name TRUMP, was developed in 1965 by the Lawrence Livermore Laboratory to solve a variety of transient and steady-state conduction heat flow problems in simple or complicated geometric configurations. The purpose of this investigation was to adapt "TRUMP" to the Naval Postgraduate School IBM/360 Model 67 computer system. Several heat conduction problems are solved with the adapted version of TRUMP and the results compare closely with the analytical solutions to these problems. The example problem inputs may be used as guides in preparing input data for future problems. Some suggestions are also given for further development of the program.

TABLE OF CONTENTS

I. INTRODUCTION ---------- 8

II. DESCRIPTION OF THE PROGRAM ---------- 10

- A. GENERAL ---------- 10
- B. USE OF COMPUTER ---------- 14
  - 1. General ---------- 14
  - 2. Block Item Description ---------- 15
  - 3. Numerical Output ---------- 16
  - 4. Evaluation of Results ---------- 16
- C. DESCRIPTIONS OF SUBROUTINES ---------- 19

III. EXAMPLE PROBLEMS ---------- 21

- A. SLAB-MELT PROBLEM ---------- 21
  - 1. Analytic Solution ---------- 21
  - 2. Trump Solution ---------- 21
- B. FINITE-SLAB PROBLEM ---------- 28
  - 1. Analytic Solution ---------- 28
  - 2. Trump Solution ---------- 28
- C. SEMI-INFINITE SLAB PROBLEM ---------- 34
  - 1. Analytic Solution ---------- 34
  - 2. Trump Solution ---------- 34
- D. CYLINDER PROBLEM ---------- 39
  - 1. Trump Solution ---------- 39
- E. SPHERE PROBLEM ---------- 45
  - 1. Analytic Solution ---------- 45
  - 2. Trump Solution ---------- 45

F. TWO DIMENSIONAL PLATE PROBLEM ---------------------- 48
  1. Analytic Solution ------------------------------ 48
  2. Trump Solution --------------------------------- 49
IV. CONCLUSIONS -- 57
COMPUTER PROGRAM: NPS IBM/360 MODEL 67 VERSION OF TRUMP ----- 58
BIBLIOGRAPHY --- 129
INITIAL DISTRIBUTION LIST ------------------------------------ 130
FORM DD 1473 --- 131

LIST OF TABLES

I. Problem Size Limits ---------------------------------- 12

II. Values of KWIT -- 18

III. Solid-Liquid Interface in the Slab Melt --------------- 26

IV. Heat Flow from Semi-Infinite Slab --------------------- 37

V. Temperature Distribution for Cylinder (Time = 50) ----------------- 43

VI. Temperature Distribution for Sphere (Time = 50) ------------------- 46

VII. Temperature Distribution for Two-Dimensional Plate in Steady-State --- 53

LIST OF ILLUSTRATIONS

Figure

1 Input Data for Slab Melt ------------------------- 24

2 Temperature Distribution in the Slab-Melt Problem - 25

3 Input Data for Finite Slab ----------------------- 30

4 Temperature Distribution in the Finite Slab Problem- 31

5 Surface Temperature History of Finite Slab Problem -- 32

6 Input Data for Semi-Infinite Slab ----------------- 35

7 Temperature Distribution in the Semi-Infinite Problem 36

8 Input Data for Cylinder --------------------------- 41

9 Input Data for Two-Dimensional Plate -------------- 50

10 Zoning of the Two-Dimensional Plate Problem -------- 52

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to my advisor, Professor Thomas E. Cooper, for his invaluable assistance in the preparation of this thesis. A special thanks is also due Dr. Richard Winfrey for suggestions given during the early part of this thesis.

I would also like to express my thanks to the Naval Postgraduate School Computer Facility staff for their guidance in the computer work.

I. INTRODUCTION

A computer program with the code name TRUMP, was developed in 1965 by Edwards [Reference 1] of the Lawrence Livermore Laboratory (LLL) to solve a wide variety of non-linear, conduction heat transfer problems. TRUMP has been extensively used and continually improved since its initial development and has been successfully adapted to a number of computer systems including the CDC/6600 (LLL), IBM/7094 (Bell Labs), UNIVAC/1108 (Battelle Northwest, Gulf General Atomics, NBS), and the IBM/360 (B.F. Goodrich, NASA Lewis Research Center, Bendix Electrodynamics, Cockerill-Ougee-Providence), to name a few. The objective of this thesis project was to adapt TRUMP to the Naval Postgraduate School (NPS) IBM/360 Model 67 computer system.

The B.F. Goodrich Corporation supplied a listing of a version of the TRUMP program that they had converted for use in their IBM/360 system. Cards were punched from this listing and a version of TRUMP was thus secured.

The B.F. Goodrich version was not complete. In addition to errors that resulted from punching 3,350 cards, the program also had an alignment error in one subroutine of the program. Also, several options that are available in the complete LLL version of TRUMP were missing in the B.F. Goodrich version.

With a great deal of assistance from the NPS Computer Facility staff, the majority of these problems have been corrected and a running version of TRUMP is now available at NPS.

No attempt is made in this thesis to give a complete description of the TRUMP program. Rather, limitations of the NPS IBM/360 version as

compared to the LLL CDC/6600 version are discussed. In addition, several example problems are solved and may be used as guides in preparing input data for future problems. For a complete description of the program, reference should be made to the TRUMP users manual [Reference 1].

II. DESCRIPTION OF THE PROGRAM

A. GENERAL

The program solves the general, non-linear, parabolic partial differential heat equation. Transient and steady-state heat conduction problems in multidimensions may be solved. Initial conditions, material properties, source and sink strength and boundary conditions may vary with time, spatial position and/or temperature.

In order to solve problems with TRUMP, the user specifies input data in "BLOCK" form to 12 input data blocks. A description of each of these blocks and their use will be given in Section B 2.

Following the definition of the problem a model must be constructed. Complex geometric shapes are divided into regions and if there is symmetry, the symmetry planes are represented by insulated surfaces in order to simplify the model. Regions may have different materials with specified properties.

The regions may be subdivided into volume elements or nodes having arbitrarily any size and shape. Nodes or volume elements must have a representative nodal point whose location may be anywhere in the node or on the surface of the node. However, in transient problems, for maximum accuracy, node shapes and nodal point locations are chosen so that the lines joining the nodal points of connected nodes are perpendicularly bisected by the connected area. In steady-state problems, due to the lack of heat capacity effects, the nodal points may be located anywhere in the node without loss of accuracy. A variety of boundary conditions may be given to the surface nodes. Constant temperature, variable temperature, constant heat flux, variable heat flux,

forced convection conditions, free convection conditions, radiation boundary conditions or combinations of the above may be treated.

Instead of having very fine zoning of the regions, rather crude zoning will give the solution with less effort. Later, the solution may be obtained for a finer subdivision of the system. Also, with the crude zoning one can easily understand which regions need finer zoning.

The accuracy of the given problem depends on how well the problem can be modeled, and the fineness of the spatial subdivisions and time steps used in the calculations.

Allowable problem size limits for the NPS IBM/360 version of TRUMP are summarized in Table 1.

TABLE I

Problem Size Limits

Item	Data Block	Parameter	Size
Materials	2	M_2	15
Reactants	3	M_3	5
Nodes	4	M_4	355
Internal thermal connections	5	M_5	950
External thermal connections	6	M_6	60
Boundary nodes	7	M_7	20
Heat generation tables	8	M_8	5
Initial conditions	9	M_4	355
Mass flow connections	10	M_{10}	50
Temp. versus time plot nodes	11	M_{11}	10
Temp. versus plot times	-	M_1	1
Remotely dependent properties	12	M_{12}	75
Table lengths	2, 3, 6, 7, 8, 10	M_9	12

In general, any consistent set of units may be used for the input data; output quantities are then given in the same unit system.

Each material, system node, and boundary node can be identified by a non-zero integer number fitting within 5-column fields on the data cards. These identifications can be in any sequence. In order to reduce the number of input cards arithmetic sequencing is suggested so that the nodes having identical descriptions can be submitted as input on only one card.

For non-symmetric, symmetric with respect to an axis, or to a center, the nodal volumes and the thermal connection areas can be calculated by a control value in the input data.

Initial conditions may be specified in BLOCK 1 of the input data. These values will be assigned to all nodes described in BLOCK 4, unless other values are specified for individual nodes in BLOCK 9.

Each internal thermal connection between nodes must be described in the input data by specifying the two node identification numbers, two connector lengths, and two interface dimensional factors. The selection of connector lengths depends on the location of the nodal point within each node. These locations may be arbitrary, but in transient problems should usually be at the geometric centers of the nodes. If the amount of the heat flow depends on surface temperatures, then they can be located on the surface.

A system that exchanges heat with its surrounding must have external temperatures specified along its boundary. In the input data, boundary node identification numbers and their temperatures, which may be constant or a function of time, must be given. The term "boundary node" is identical in meaning to an external temperature.

Surface temperatures can be specified by using boundary nodes with specified temperatures and connecting the surface nodes to a boundary node with a very large convection coefficient. In order to reach a high degree of accuracy, surface nodes should be selected as either zero-volume nodes, or very thin nodes. When a large number of surface nodes are connected to the same boundary node and the heat transfer coefficients are not tabulated, the number of external connections can be reduced by replacing them with connections to a single zero-volume node that is connected to the boundary node with a very large convection coefficient.

In order to generate steady-state solutions, TRUMP must first go through a set of transient calculations. The steady-state solution is obtained by allowing enough problem time to take the transient to its conclusion. However, if only steady-state results are desired, the control on the average temperature change during a time step (TVARY) can be made very large. This allows the program to neglect transient stability limitations on its choice of time steps resulting in a minimum expenditure of computer time on transient calculations.

B. USE OF COMPUTER

1. General

The data deck must include a problem name card, any number of block number cards with their input cards, and a data end card. The first card of each data deck must be a problem name card. The symbol "*" must be in column 1, any identification of the problem in columns 2 through 71 and 73 through 80.

In the program there are 12 allowed input data blocks, with block numbers from 1 through 12. Each block must begin with a block number card, and except for BLOCKS 1 and 11, must end with a blank card. The block number card must begin with the word "BLOCK" in columns 1 through 5, and the block number, from 01 through 12, in columns 6 and 7.

Data BLOCK 1 must follow the problem name card; all other data blocks may be placed in any order.

The last card of each data deck must be a data end card, with the word "ENDED" in columns 1 through 5, and "-1" in columns 6 and 7.

Additional cards containing comments without "*" in column 1 may precede the name card and will appear on the printout. Any desired block description in columns 9 through 80 may be made on the block number card. This will also appear on the printout.

2. Block Item Descriptions

Basically, one should refer to the TRUMP users manual [Reference 1] for format details and the original descriptions of the input values. In this section the main difference between the IBM/360, Model 67 version of TRUMP and the LLL, CDC/6600 version are discussed.

In BLOCK 1;

MCYC	Maximum allowed number of time steps. If zero, negative, or unspecified will be 30,000.
MSEC	Maximum allowed machine time, in seconds. If zero, negative, or unspecified will be 30,000.
NPUNCH	Indicates that the final values of temperatures of nodes, weight fractions of reactants, and constant heat generation rates of nodes will be punched. Since punched

	output is not obtained in this version of TRUMP, the value of NPUNCH should not be used.
IRITE	Number of time steps between data output on unit ITAPE of problem time and table of node numbers and node temperatures, in addition to output of first, second, and last time steps. Only used in TRUMP/6600/DS version. This control is not in use in this version.
ITAPE	Output unit on which output data will be written as described in IRITE. This input value is not in use for IBM/360 Model 67 version of TRUMP.
TIMEP	Problem time interval between data output, in addition to output on first, second, and last time steps. This input control value is also not in use for this version of TRUMP.

In addition to the above differences, the data transfer option is not in use in this version of TRUMP.

3. Numerical Output

The amount of output data is controlled by the value of KDATA. However, for every problem, all input data and several other quantities such as node volumes and connection areas are written out. The results of the first, second, and last time steps are always written out. In addition, output data are written out at problem time step intervals of IPRINT. For the notations used on the printouts one should refer to TRUMP users manual [Reference 1].

4. Evaluation of Results

Any desired end to the problem may be made with the specified input values in BLOCK 1. These may be maximum problem time, TIMAX,

maximum temperature, TMAX, minimum temperature, TMIN, maximum number of time steps, MCYC, or maximum number of seconds of machine time, MSEC. On the printouts, the value of the problem-end sentinel, KWIT, will appear. Table II shows the values of KWIT, and the descriptions of these values.

TABLE II

Values of KWIT

KWIT	Cause of Problem Ending
1	The problem time, SUMTIM, reached TIMAX
2	A temperature exceeding TMAX
3	A temperature less than TMIN
4	Steady-state
5	Data
6	Relates to the tape usage
7	The number of time steps reached MCYC
8	Machine time reached MSEC
9	BLOCK 2 or BLOCK 4 missing
10	Convergence failure
11	Items exceeded the size limits
12	Table lengths exceeded the size

C. DESCRIPTIONS OF SUBROUTINES

THERM	Subroutine for material properties and node descriptions. BLOCKS 2, 4, and 12 are included.
TALLY	Subroutine for initialization, totaling, and checking results, converting regular to special nodes, testing for ending the problem. BLOCKS 1 and 9 are included.
CHEM	Subroutine for chemical reaction. BLOCK 3 is included.
SPECK	Subroutine for heat flow between special nodes and other nodes or boundary nodes. Iterates to solve set of implicit difference equations when special nodes connected to each other.
FINK	Subroutine for heat flow between nodes by conduction, convection, and radiation. Treats all nodes as regular nodes. BLOCK 5 is included.
GEN	Subroutine for internal heat generation. BLOCK 8 is included.
SURE	Subroutine for heat flow between surface nodes, boundary nodes by radiation, free and forced convection treats all surface nodes as regular nodes. BLOCKS 6 and 7 are included.
FLOW	Subroutine for mass flow between nodes. Treats all nodes as regular nodes. BLOCK 10 is included.
SEEK1	Subroutine for finding an identification number of a material, reactant, node or boundary node in a block item list.
SEEK2	Subroutine for finding the identification numbers of a pair of nodes in a block item list.

CLOCK,CLOCK1	Subroutines for initialization before first time step.
PLOT	Subroutine for making CRT plots and tables of temperature versus node location, temperature versus time, and node location versus time. BLOCK 11 is included.
PATCH	Subroutine for converting a number with a decimal point to floating number, otherwise substituting a specified floating point number.
REFER	Subroutine for finding array subscripts of materials, reactants, nodes or boundary nodes referred to by identification number in another data block, and writing out diagnostic statements whenever one can not be found.
SRCON	Subroutine which is related to PATCH. Converts a number having decimal point to a floating point number.

III. EXAMPLE PROBLEMS

A. SLAB-MELT PROBLEM

A semi-infinite slab ($-\infty < x < \infty$, $-\infty < y < \infty$, $0 \leqslant z < \infty$), has thermal conductivity 1.0 cal/sec-°C, density 1.0 g/cm^3, heat capacity 1.0 cal/sec-°C, a melting point of 50.0°C, and a latent heat of fusion of 25.0 cal/g. The initial temperature is 100.0°C, and for times $t > 0$, the surface, $z = 0$, is maintained at 0.0°C. TRUMP is used to calculate the temperature distribution in the slab, and the motion of the solid-liquid interface, Z_{int}, for times up to 200 sec. The results are compared with the exact solution.

1. Analytic Solution

From Reference 3 analytic results are:

$$T_1 = 112.4 \operatorname{erf}\left(\frac{Z}{2\sqrt{t}}\right) \quad \text{for} \quad 0 \leqslant T < 50^\circ C$$

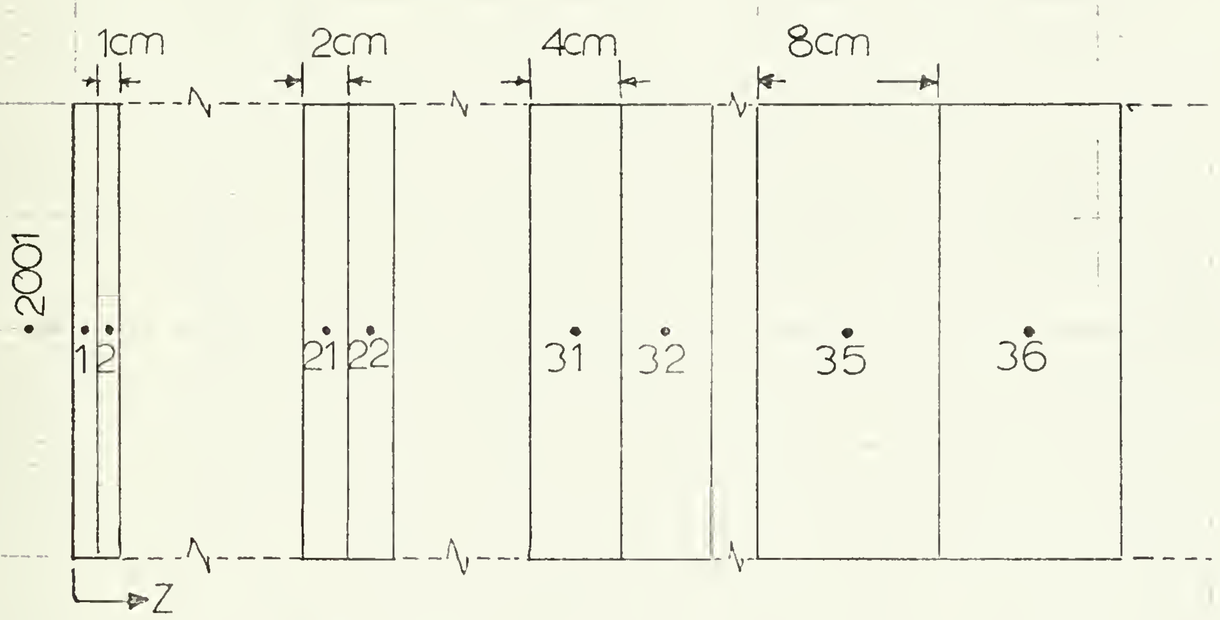
$$T_2 = 9.9 + 90.1 \operatorname{erf}\left(\frac{Z}{2\sqrt{t}}\right) \quad \text{for} \quad 50 \leqslant T \leqslant 100^\circ C$$

$$Z_{int} = 0.835 \cdot \sqrt{t} \quad \text{for } T = 50^\circ C$$

2. Trump Solution

The input data for this problem are shown in Figure 1. In order to have output at every 10 cycles, in addition to output on the first, second, and last time steps, IPRINT is selected as 10 in BLOCK 1. For temperature accuracy, TVARY is 1.0°C and the problem is ended at 200 sec by setting TIMAX equal to 200.0. TONE is 100.0 since the initial temperature of the slab is 100.0°C. However, since the zero-volume surface node, node -1, is at 0.0°C for $t > 0$ its temperature is specified in BLOCK 9 by setting the value of TT equal to 0.0. In BLOCK 2, the

material name, AMAT, and material identification number are arbitrarily selected at SMELT and 50, respectively. Other slab thermal properties such as density, heat capacity, thermal conductivity, temperature at which the latent heat is released and the latent heat of fusion are also specified in BLOCK 2. In order to give node descriptions in BLOCK 4 the slab is divided into nodes as shown below.



A total field depth of 100.0 cm is chosen so that at 200 sec no appreciable temperature change will take place at the maximum value of Z. In this figure, the node identification numbers, and the node thicknesses are shown. The nodal points of all nodes are at the node centers. In BLOCK 5, internal connection node numbers, lengths of heat conduction paths from the nodal points to the connected interfaces, and the area of the connected interfaces are defined. In BLOCK 6, a zero-volume surface node, node -1, is connected to a boundary node, node 2001, with a heat transfer coefficient of 1.0 x 10^8cal/sec-cm^2-^{o}C,

and the external surface area of the surface node is selected as unity. In BLOCK 7, a constant boundary node temperature is defined as 0.0°C. Computer running time for this problem was 65.52 sec.

```
*     SLAB  MELT  PROBLEM.  SEPT.  1971
NOTE.    H(M) = 25 CAL/G, T(M) = 50.0, TIME 0-200 SEC.
NOTE.    THIS PROBLEM DOES NOT USE GEN, CHEM, OR FLOW SUBROUTINES.
BLOCK 1
   10
                               1.0                 200.0
 100.0
BLOCK 2  MATERIAL NAMES, NUMBERS, CHEMICAL COMPONENTS, AND THERMAL PROPERTIES.
SMELT   50                     1.0       1.0       1.0       50.0      25.0

BLOCK 4   NODE NUMBERS, MATERIALS, TYPES, AND DIMENSIONS.
   -1             50
    1   19     1   50           1.0       1.0       1.0
   21    9     1   50           2.0       1.0       1.0
   31    4     1   50           4.0       1.0       1.0
   36    4     1   50           8.0       1.0       1.0

BLOCK 5  INTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1    1                     0.5       1.0       1.0
    1    2  18   1  1  0.5     0.5       1.0       1.0
   20   21           0.5       1.0       1.0       1.0
   21   22   8   1  1  1.0     1.0       1.0       1.0
   30   31           1.0       2.0       1.0       1.0
   31   32   3   1  1  2.0     2.0       1.0       1.0
   35   36           2.0       4.0       1.0       1.0
   36   37   3   1  1  4.0     4.0       1.0       1.0

BLOCK 6  EXTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1 2001                     1.0       1.0       1.0    E8

BLOCK 7  BOUNDARY NODE TEMPERATURES.
 2001                0.0

BLOCK 9   INITIAL CONDITIONS.
   -1         1      0.0

ENDED-1  LAST CARD OF DATA DECK.  MORE DATA DECKS MAY FOLLOW.
```

FIG. 1 INPUT DATA FOR SLAB MELT

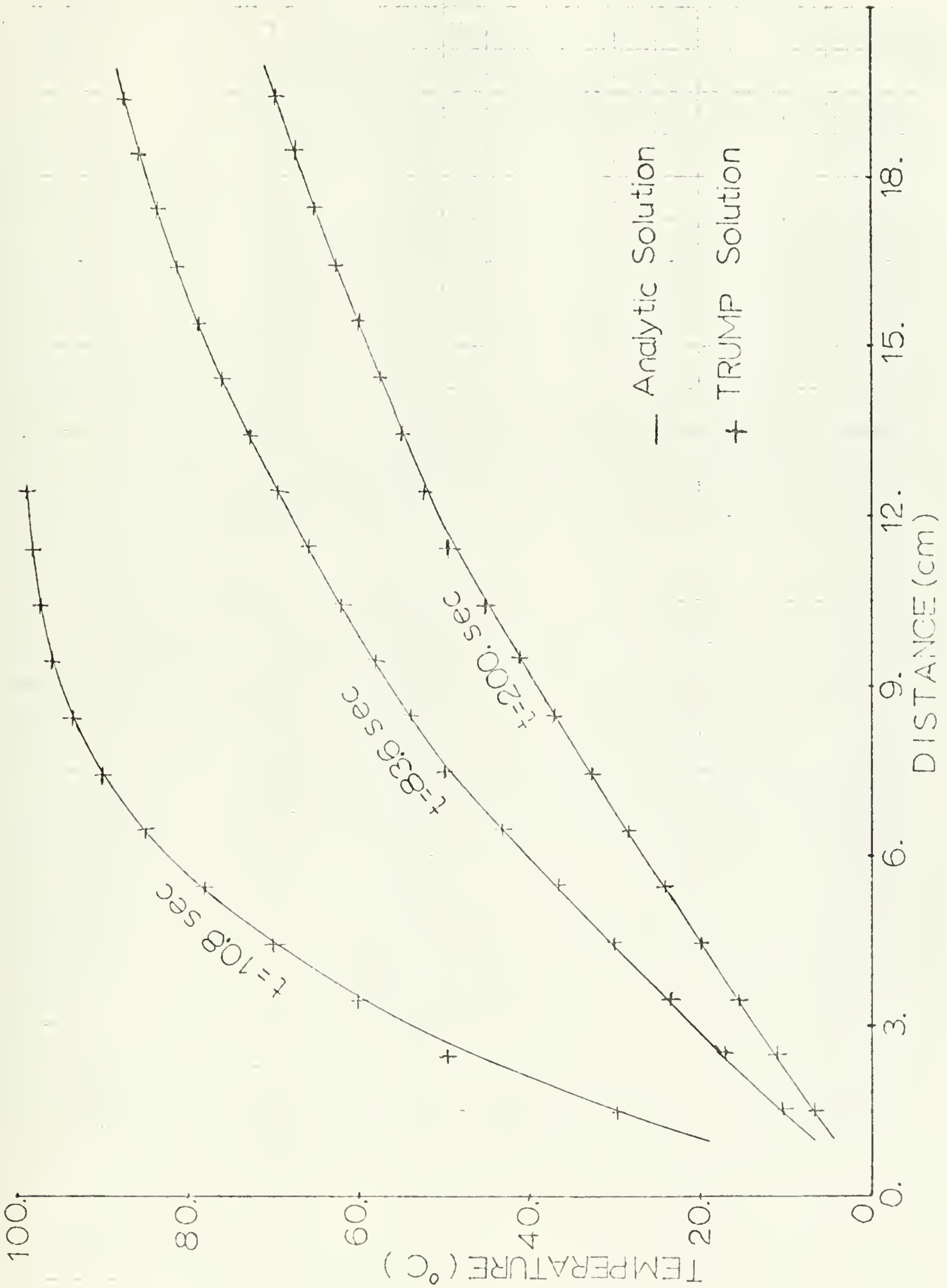


FIG. 2 TEMPERATURE DISTRIBUTION IN THE SLAB-MELT PROBLEM

TABLE III

SOLID-LIQUID INTERFACE IN THE SLAB MELT

Time (sec)	Z_{int} (Analytic)	Z_{int} (Trump)
3.56	1.57	1.37
4.19	1.71	1.77
4.77	1.82	2.00
5.92	2.03	2.00
8.13	2.38	2.04
9.67	2.60	2.46
10.81	2.74	2.86
12.08	2.90	3.00
15.90	3.33	3.02
18.43	3.58	3.44
20.13	3.75	3.84
22.21	3.94	4.00
28.24	4.44	4.16
30.91	4.64	4.56
33.10	4.80	4.97
38.20	5.16	5.00
44.21	5.55	5.37
47.20	5.74	5.77
50.42	5.93	6.00
60.16	6.48	6.23
64.03	6.68	6.63
67.34	6.85	7.00
78.78	7.41	7.13
83.62	7.64	7.54
87.67	7.82	7.94
106.12	8.60	8.46
110.92	8.79	8.87
120.92	9.18	9.00
131.54	9.58	9.42
137.10	9.78	9.82
146.27	10.10	10.00
160.05	10.56	10.39
166.34	10.77	10.79
175.46	11.06	11.00
191.54	11.56	11.37
198.54	11.77	11.77
200.00	11.81	11.86

```
TRUMP OUTPUT DATA                                                                                   DATA DECK    1
  * SLAB MELT.  SAMPLE PROBLEM FOR TRUMP REPORT.
=====================================================================================================================

   PRINTOUT     CYCLE    TOO FAST  TOO SLOW     KWIT      DELTMX           SMALL             TVARY         NUTS
      52         493        0         0          1     4.57143E 00      1.60000E-02       1.00000E 00       1

    TOTAL TIME      TIME STEP       HEAT FLOW      TEMP FROM FLUX   FLUX RATE       TEMP RATE
   2.00000E 02    1.59607E-01   -1.78806E 03     -1.78806E 01    -8.94029E 00    -8.94029E-02

    AVG TEMP      HEAT CAPACITY   HEAT CONTENT     GEN RATE          HEAT GEN       TEMP FROM GEN
   8.50838E 01    1.000000D 02    1.07118E 04     0.0                0.0            0.0
=====================================================================================================================

 NODE         TEMP          DT            DDT        GE N RATE        W            H              F            CURE AT 280 F
  -1       0.4457D-07  -0.1321D-10   -0.2939D-10     0.0         0.4457E-31   -0.2500E-22    0.3657E-02     0.2492E 00
   1       0.2229D 01  -0.6736D-03   -0.1498D-02     0.0         0.2229E 01   -0.1228E 03   -0.1228E 03     0.3077E 00
   2       0.6682D 01  -0.1968D-02   -0.4379D-02     0.0         0.6682E 01   -0.1183E 03   -0.1183E 03     0.4724E 00
   3       0.1112D 02  -0.3106D-02   -0.6909D-02     0.0         0.1112E 02   -0.1139E 03   -0.1139E 03     0.6936E 00
   4       0.1554D 02  -0.3996D-02   -0.8893D-02     0.0         0.1554E 02   -0.1095E 03   -0.1094E 03     0.9619E 00
   5       0.1994D 02  -0.4570D-02   -0.1017D-01     0.0         0.1994E 02   -0.1051E 03   -0.1050E 03     0.1270E 01
   6       0.2431D 02  -0.4786D-02   -0.1065D-01     0.0         0.2431E 02   -0.1007E 03   -0.1007E 03     0.1610E 01
   7       0.2865D 02  -0.4635D-02   -0.1031D-01     0.0         0.2865E 02   -0.9635E 02   -0.9634E 02     0.1977E 01
   8       0.3296D 02  -0.4143D-02   -0.9216D-02     0.0         0.3296E 02   -0.9204E 02   -0.9203E 02     0.2365E 01
   9       0.3724D 02  -0.3347D-02   -0.7445D-02     0.0         0.3724E 02   -0.8776E 02   -0.8775E 02     0.2769E 01
  10       0.4150D 02  -0.2281D-02   -0.5074D-02     0.0         0.4150E 02   -0.8349E 02   -0.8349E 02     0.3183E 01
  11       0.4575D 02  -0.6179D-03   -0.1375D-02     0.0         0.4575E 02   -0.7924E 02   -0.7924E 02     0.3602E 01
  12       0.5000D 02   0.5829D-14    0.1297D-37     0.0         0.5339E 02   -0.7161E 02   -0.7161E 02     0.4022E 01
  13       0.5267D 02  -0.1947D-02   -0.4332D-02     0.0         0.7767E 02   -0.4732E 02   -0.4732E 02     0.4437E 01
  14       0.5534D 02  -0.4957D-02   -0.1103D-01     0.0         0.8034E 02   -0.4466E 02   -0.4466E 02     0.4853E 01
  15       0.5797D 02  -0.7405D-02   -0.1647D-01     0.0         0.8297E 02   -0.4203E 02   -0.4202E 02     0.5266E 01
  16       0.6056D 02  -0.9629D-02   -0.2142D-01     0.0         0.8556E 02   -0.3944E 02   -0.3944E 02     0.5675E 01
  17       0.6308D 02  -0.1160D-01   -0.2581D-01     0.0         0.8808E 02   -0.3692E 02   -0.3691E 02     0.6075E 01
  18       0.6554D 02  -0.1329D-01   -0.2957D-01     0.0         0.9054E 02   -0.3446E 02   -0.3446E 02     0.6466E 01
  19       0.6791D 02  -0.1466D-01   -0.3260D-01     0.0         0.9291E 02   -0.3209E 02   -0.3209E 02     0.6846E 01
  20       0.7019D 02  -0.1569D-01   -0.3490D-01     0.0         0.9519E 02   -0.2981E 02   -0.2981E 02     0.7213E 01
  21       0.7346D 02  -0.1676D-01   -0.3729D-01     0.0         0.1969E 03   -0.5308E 02   -0.5308E 02     0.7750E 01
  22       0.7740D 02  -0.1708D-01   -0.3800D-01     0.0         0.2048E 03   -0.4520E 02   -0.4520E 02     0.8392E 01
  23       0.8091D 02  -0.1665D-01   -0.3704D-01     0.0         0.2118E 03   -0.3817E 02   -0.3817E 02     0.8965E 01
  24       0.8401D 02  -0.1578D-01   -0.3511D-01     0.0         0.2180E 03   -0.3198E 02   -0.3198E 02     0.9468E 01
  25       0.8671D 02  -0.1468D-01   -0.3266D-01     0.0         0.2234E 03   -0.2657E 02   -0.2657E 02     0.9900E 01
  26       0.8905D 02  -0.1346D-01   -0.2994D-01     0.0         0.2281E 03   -0.2191E 02   -0.2191E 02     0.1027E 02
  27       0.9104D 02  -0.1218D-01   -0.2711D-01     0.0         0.2321E 03   -0.1791E 02   -0.1791E 02     0.1057E 02
  28       0.9273D 02  -0.1089D-01   -0.2422D-01     0.0         0.2355E 03   -0.1453E 02   -0.1453E 02     0.1082E 02
  29       0.9415D 02  -0.9604D-02   -0.2136D-01     0.0         0.2383E 03   -0.1169E 02   -0.1169E 02     0.1103E 02
  30       0.9533D 02  -0.8361D-02   -0.1860D-01     0.0         0.2407E 03   -0.9340E 01   -0.9340E 01     0.1119E 02
  31       0.9678D 02  -0.6558D-02   -0.1459D-01     0.0         0.4871E 03   -0.1287E 02   -0.1287E 02     0.1138E 02
  32       0.9806D 02  -0.4678D-02   -0.1041D-01     0.0         0.4922E 03   -0.7755E 01   -0.7755E 01     0.1153E 02
  33       0.9887D 02  -0.3176D-02   -0.7065D-02     0.0         0.4955E 03   -0.4515E 01   -0.4515E 01     0.1162E 02
  34       0.9936D 02  -0.2063D-02   -0.4590D-02     0.0         0.4974E 03   -0.2550E 01   -0.2550E 01     0.1167E 02
  35       0.9965D 02  -0.1292D-02   -0.2874D-02     0.0         0.4986E 03   -0.1414E 01   -0.1414E 01     0.1169E 02
  36       0.9988D 02  -0.5411D-03   -0.1204D-02     0.0         0.9990E 03   -0.9774E 00   -0.9774E 00     0.1171E 02
  37       0.9997D 02  -0.1663D-03   -0.3700D-03     0.0         0.9997E 03   -0.2500E 00   -0.2500E 00     0.1172E 02
  38       0.9999D 02  -0.4538D-04   -0.1009D-03     0.0         0.9999E 03   -0.5805E-01   -0.5805E-01     0.1172E 02
  39       0.1000D 03  -0.1119D-04   -0.2490D-04     0.0         0.1000E 04   -0.1238E-01   -0.1238E-01     0.1172E 0
  40       0.1000D 03  -0.2957D-05   -0.6578D-05     0.0         0.1000E 04   -0.2833E-02   -0.2833E-02     0.1172E
=====================================================================================================================
```

B. FINITE-SLAB PROBLEM (suggested by Professor Paul Pucci)

A large, 1.0 inch thick steel plate, initially at 80.0°F, is suddenly submerged in a fluid at temperature 680.0°F. TRUMP is used to calculate the transient temperature distribution in the plate assuming a constant surface heat transfer coefficient of 1248.0 BTU/hr-ft^{2}-$^{\circ}$F. The thermophysical properties of the steel were taken as: thermal conductivity 26.0 BTU/hr-ft-$^{\circ}$F, density 487.0 lbm/ft^{3}, and specific heat 0.133 BTU/lbm-$^{\circ}$F.

1. Analytic Solution

$$\frac{T - T_{\infty}}{T_i - T_{\infty}} = 2\sum_{n=1}^{\infty} \frac{\mathrm{Sin}\,\lambda_n L}{\lambda_n L + \mathrm{Sin}\,\lambda_n L\,\mathrm{Cos}\,\lambda_n L}\, e^{-(\lambda_n L)^2 Fo}\, \mathrm{Cos}\left(\lambda_n L \frac{x}{L}\right)$$

Applying the Biot number, $Bi = \frac{hL}{k} = 2.0$ to the eigen value problem of $\mathrm{Cot}(\lambda_n L) = \frac{\lambda_n L}{Bi}$, the two eigen values $\lambda_1 L = 1.0769$ and $\lambda_2 L = 3.6436$ can be found. After substituting the appropriate value of the Fourier number, $Fo = \frac{\alpha t}{L^2} = 230.4t$ in the above series solution and considering only the first two terms, the result becomes:

$$(x,t) = 680 - 1200\left[0.58923153\, e^{-267.2t}\, \mathrm{Cos}\left(1.0769\frac{x}{L}\right) - 0.11836085 e^{-3058.75t}\, \mathrm{Cos}\left(3.6436\frac{x}{L}\right)\right]$$

2. Trump Solution

The input data for this problem are shown in Figure 3. For unit consistency, lengths in inches are converted to feet and applied to the problem. Since the plate has symmetry with respect to the mid-plane, this plane is taken as an insulated plane in the TRUMP solution. So, in BLOCK 1 symmetry type indicator, KD, is applied as 1 for the non-symmetric case, and the total thickness of the steel plate is considered as 0.5 inches. Temperature scale indicator, KT, is selected as 3 because of the Fahrenheit scale. For temperature accuracy, the desired

maximum temperature change in each time step, TVARY, is selected as 1.0°F. Maximum problem time, TIMAX, is given as 1.0 hr. TONE is set equal to 80.0 making the initial temperature of all nodes equal to 80.0°F. In BLOCK 2, material name and material number are given as, FSLAB and 50, respectively. Also, the steel properties such as density, specific heat, thermal conductivity are defined. In BLOCK 4 node numbers, materials of nodes and their dimensions are given with respect to the equally spaced 10 nodes for the 0.5 inches of thickness of the slab. In BLOCK 6, the zero-volume surface node, node -1, is connected to a boundary node, node 2001, with a surface heat transfer coefficient of 1248.0 BTU/hr-ft^2-°F. In BLOCK 7, a boundary node temperature of 680.0°F is given. The computer running time for this problem was 24.46 sec.

In order to see the effect of finer zoning on the results, the same problem was also solved using 20 and 40 nodes. However, the results were all within 1°F accuracy and not much improvement was obtained with the finer grid sizes. Computer running time, on the other hand, was increased from 24.46 sec. to 53.76 sec. and to 172.64 sec. for 10, 20, and 40 nodes, respectively.

```
* TEMPERATURE DISTRIBUTION IN A FINITE SLAB. 10/5/1971.
BLOCK 1   PROBLEM  CONTOLS.
   10
    1    3                            1.                  1.
       80.
BLOCK 2  MATERIAL NAMES, NUMBERS, CHEMICAL COMPONENTS, AND THERMAL PROPERTIES.
FSLAB   50                          487.      .133       26.

BLOCK 4   NODE NUMBERS, MATERIALS, TYPES, AND DIMENSIONS.
   -1             50
    1    0    1   50          .004166667        1.        1.

BLOCK 5  INTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1    1                    .002083333        1.        1.
    1    2  8  1  1 .002083333.002083333        1.        1.

BLOCK 6  EXTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1 2001                            1.        1.1.248   E3

BLOCK 7  BOUNDARY NODE TEMPERATURES.
 2001                     680.

ENDED-1  LAST CARD OF DATA DECK.  MORE DATA DECKS MAY FOLLOW.
```

FIG. 3 INPUT DATA FOR FINITE SLAB

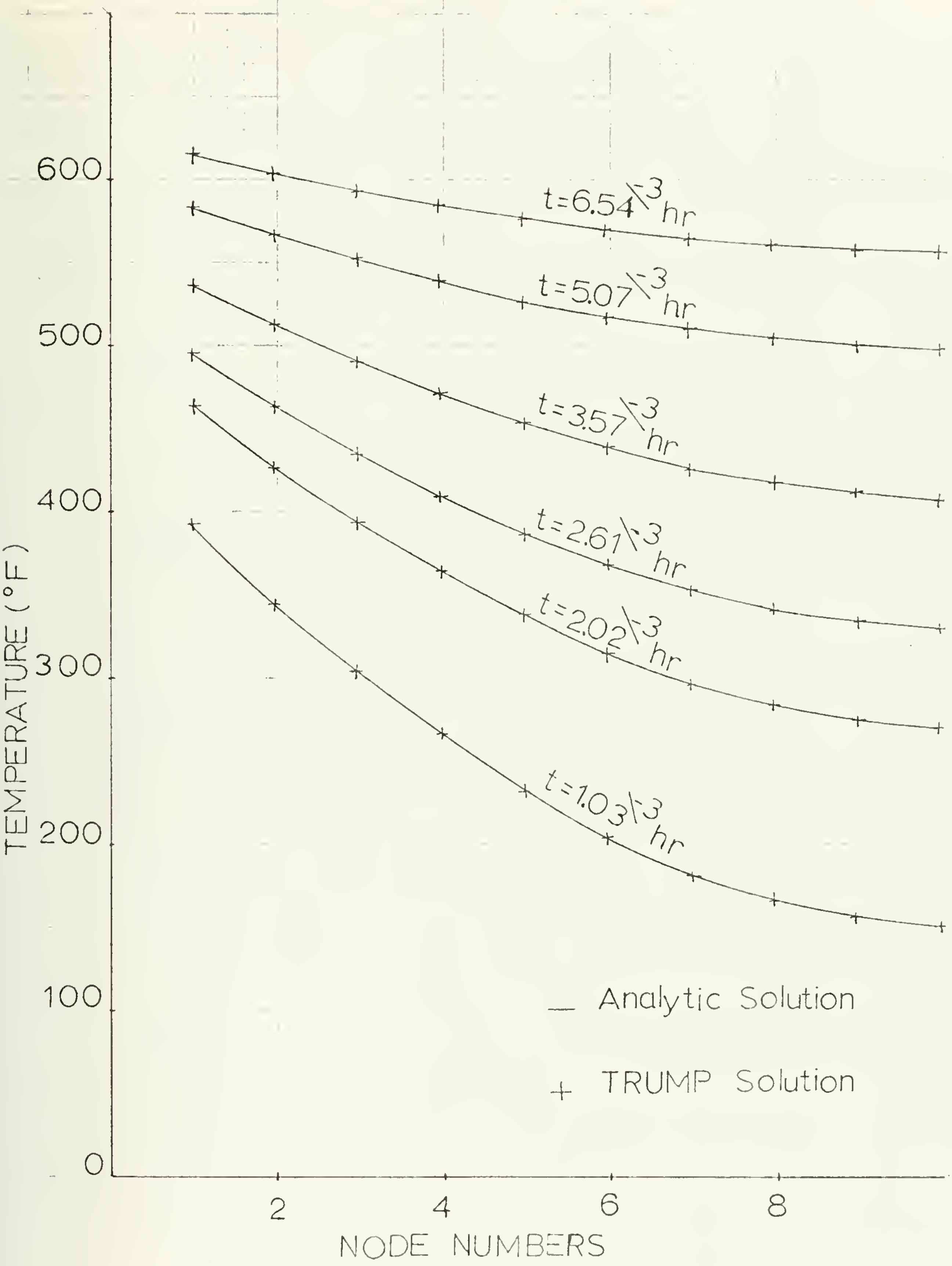


FIG. 4 TEMPERATURE DISTRIBUTION IN THE FINITE SLAB PROBLEM

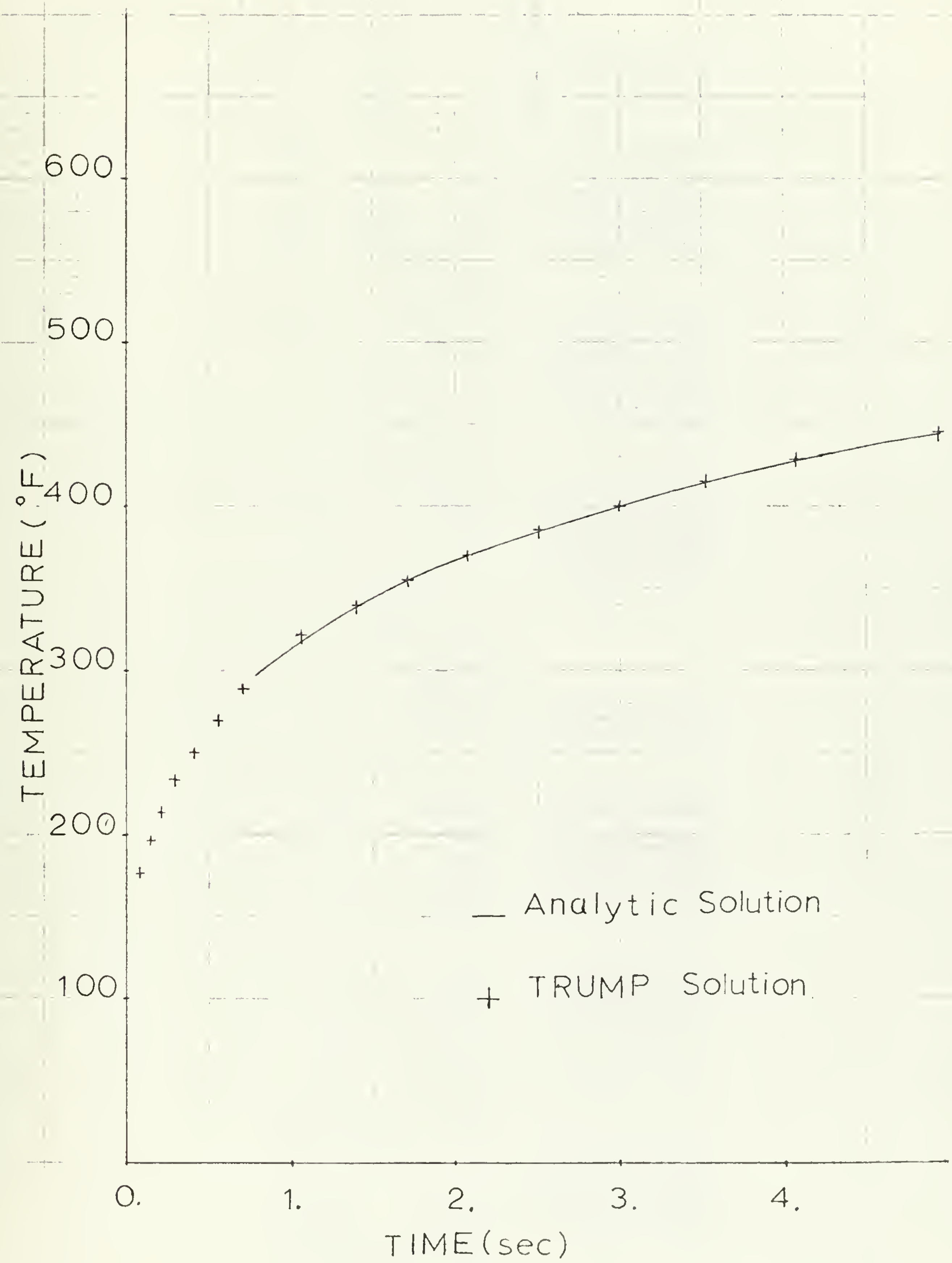


FIG. 5 SURFACE TEMPERATURE HISTORY OF FINITE SLAB PROBLEM

```
TRUMP OUTPUT DATA                                                                                   DATA DECK     1

* TEMPERATURE DISTRIBUTION IN A FINITE SLAB. 10/5/1971.
=======================================================================================================================

  PRINTOUT     CYCLE    TOO FAST   TOO SLOW      KWIT       DELTMX            SMALL              TVARY        NUTS
     91         889         0          0           1      1.00000E 12       1.44166E-07        1.00000E 00      0

    TOTAL TIME      TIME STEP       HEAT FLOW      TEMP FROM FLUX    FLUX RATE       TEMP RATE
   1.00000E 00     2.04880E-01     1.61419E 03      5.98117E 02     1.61419E 03     5.98117E 02

    AVG TEMP      HEAT CAPACITY    HEAT CONTENT       GEN RATE           HEAT GEN        TEMP FROM GEN
   6.80007E 02     2.69879D 00     1.83519E 03      0.0                0.0              0.0
=======================================================================================================================

 NODE        TEMP          DT           DDT         GE N RATE           W            H             F         CURE AT 280 F
   -1     0.6800D 03   -0.4474D-04   -0.2183D-03     0.0           0.4404E-19    0.3886E-19   -0.3576E 00      0.0
    1     0.6800D 03   -0.4885D-04   -0.2384D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
    2     0.6800D 03   -0.5448D-04   -0.2659D-03     0.0           0.1835E 03    0.1619E 03    0.1615E 03      0.0
    3     0.6800D 03   -0.5971D-04   -0.2914D-03     0.0           0.1835E 03    0.1619E 03    0.1615E 03      0.0
    4     0.6800D 03   -0.6405D-04   -0.3126D-03     0.0           0.1835E 03    0.1619E 03    0.1615E 03      0.0
    5     0.6800D 03   -0.6782D-04   -0.3310D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
    6     0.6800D 03   -0.7105D-04   -0.3468D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
    7     0.6800D 03   -0.7262D-04   -0.3545D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
    8     0.6800D 03   -0.7430D-04   -0.3626D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
    9     0.6800D 03   -0.7386D-04   -0.3605D-03     0.0           0.1835E 03    0.1619E 03    0.1614E 03      0.0
   10     0.6800D 03   -0.7484D-04   -0.3653D-03     0.0           0.1835E 03    0.1619E 03    0.1617E 03      0.0
=======================================================================================================================
```

C. SEMI-INFINITE SLAB PROBLEM

A semi-infinite slab is initially at a uniform temperature of 100.0°C. At time $t = 0$, the surface temperature is suddenly lowered and maintained at a temperature of 0.0°C. The slab has a thermal conductivity of 1.0 cal/sec cm-°C, density of 1.0 $\mathrm{g/cm^3}$, and a heat capacity of 1.0 cal/sec-°C. The temperature distibution in the slab, and the total heat flow from the slab are calculated as a function of time for times up to 200 sec using TRUMP, and the results are compared with the analytical solutions.

1. Analytical Solution

$$T(x,t) = 100 \ \mathrm{erf} \left(\frac{x^2}{4t} \right)^{\frac{1}{2}}$$

$$Q_{tot} = -112.84 \sqrt{t}$$

2. Trump Solution

The input data form is shown in Figure 6. In BLOCK 1, problem controls such as problem time, TIMAX = 200.00 sec, maximum temperature change in each time step, TVARY = 1.0°C, and the initial temperature of all nodes, TONE = 100.0°C are given. In BLOCKS 2, 4, 5, 6, 7 and 9, the thermal properties, zoning procedure, identification numbers, internal and external connection descriptions, boundary node temperatures, and initial conditions are defined identically as in the slab-melt problem. Computer running time for this problem was 27.05 sec.

```
* TRANSIENT HEAT FLOW IN SEMI-INFINITE SOLID. 10/5/1971.
NOTE.    TIME 0-200 SEC.
BLOCK 1   PROBLEM  CONTROLS.
   10
                               1.0                 200.0
 100.0
BLOCK 2  MATERIAL NAMES, NUMBERS, CHEMICAL COMPONENTS, AND THERMAL PROPERTIES.
SLAB    50                     1.0       1.0       1.0

BLOCK 4   NODE NUMBERS, MATERIALS, TYPES, AND DIMENSIONS.
   -1             50
    1   19    1   50           1.0       1.0       1.0
   21    9    1   50           2.0       1.0       1.0
   31    4    1   50           4.0       1.0       1.0
   36    4    1   50           8.0       1.0       1.0

BLOCK 5  INTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1    1                     0.5       1.0       1.0
    1    2 18  1  1  0.5       0.5       1.0       1.0
   20   21           0.5       1.0       1.0       1.0
   21   22  8  1  1  1.0       1.0       1.0       1.0
   30   31           1.0       2.0       1.0       1.0
   31   32  3  1  1  2.0       2.0       1.0       1.0
   35   36           2.0       4.0       1.0       1.0
   36   37  3  1  1  4.0       4.0       1.0       1.0

BLOCK 6  EXTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1 2001                     1.0       1.0       1.0   E8

BLOCK 7  BOUNDARY NODE TEMPERATURES.
 2001                0.0

BLOCK 9   INITIAL CONDITIONS.
   -1         1      0.0

ENDED-1  LAST CARD OF DATA DECK.  MORE DATA DECKS MAY FOLLOW.
```

FIG. 6 INPUT DATA FOR SEMI-INFINITE SLAB

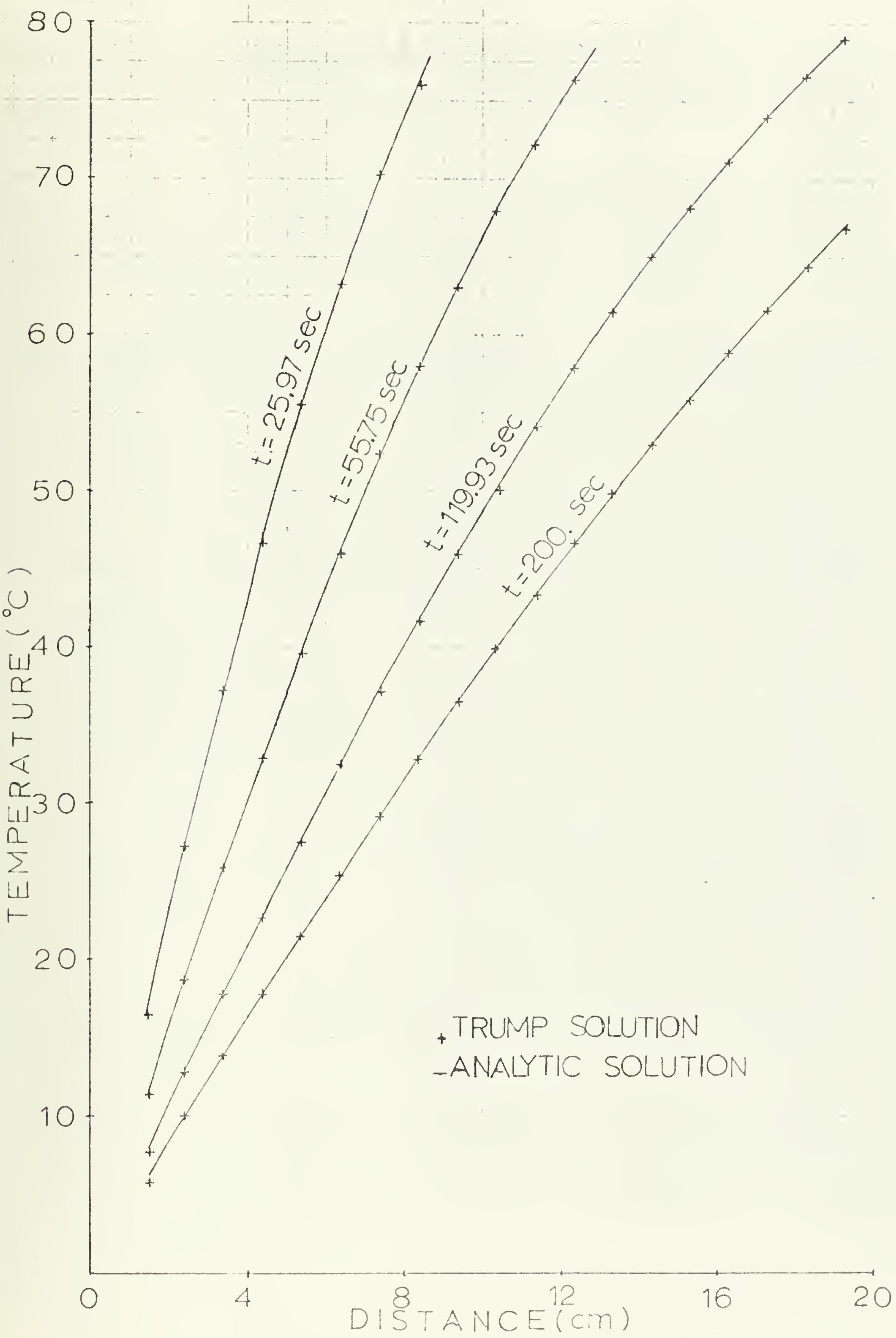


FIG. 7 TEMPERATURE DISTRIBUTION IN THE SEMI-INFINITE SLAB PROBLEM

TABLE IV

HEAT FLOW FROM SEMI-INFINITE SLAB

Time (sec)	Q_{tot} (Trump)	Q_{tot} (Analytic)
1.26	-119.42	-126.56
1.81	-146.12	-151.86
2.65	-178.93	-183.56
3.85	-217.69	-221.43
5.63	-264.69	-267.69
8.21	-320.90	-323.39
12.06	-389.63	-391.79
17.70	-472.82	-474.75
25.97	-573.36	-575.00
38.02	-694.20	-695.73
55.75	-841.12	-842.50
81.76	-1019.00	-1020.29
119.93	-1234.40	-1235.71
175.44	-1493.20	-1494.59
200.00	-1594.30	-1595.77

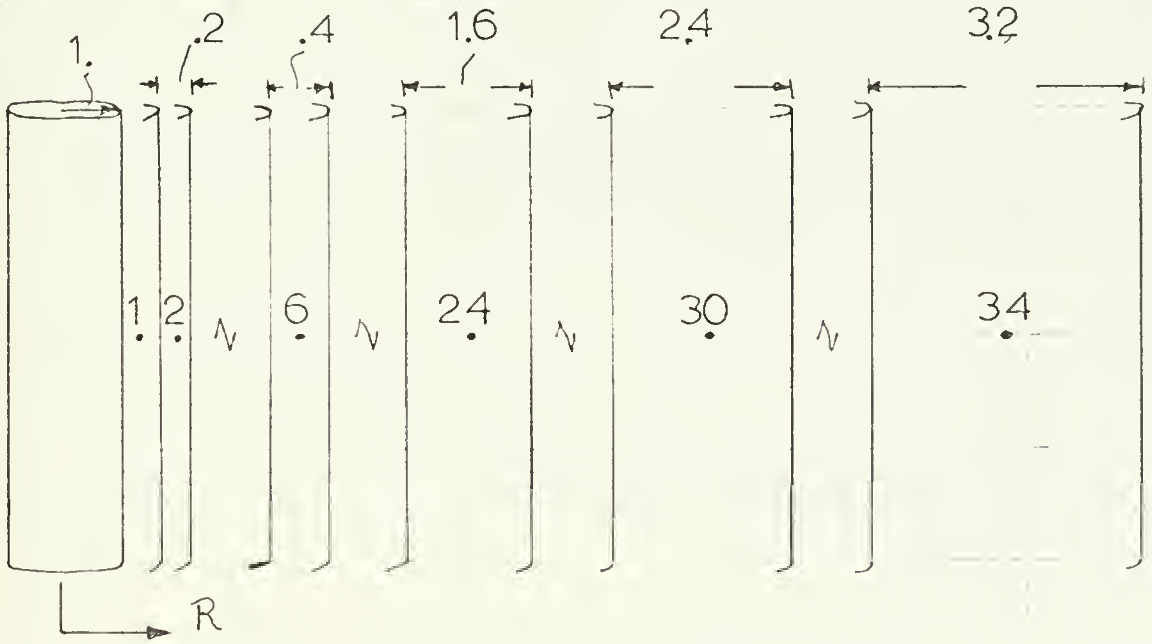
```
TRUMP OUTPUT DATA                                                                          DATA DECK    1

* TRANSIENT HEAT FLOW IN SEMI-INFINITE SOLID. 10/5/1971.
======================================================================================================

  PRINTOUT    CYCLE    TOO FAST   TOO SLOW     KWIT      DELTMX          SMALL            TVARY       NUTS
     24        214         0           0         1     1.82857E 01    1.60000E-02     1.00000E 00    13

   TOTAL TIME      TIME STEP       HEAT FLOW     TEMP FROM FLUX   FLUX RATE        TEMP RATE
  2.00000E 02     3.34782E 00    -1.59431E 03    -1.59431E 01    -7.97156E 00    -7.97155E-02

   AVG TEMP      HEAT CAPACITY    HEAT CONTENT     GEN RATE          HEAT GEN        TEMP FROM GEN
  8.40531E 01     1.00000D 02     8.40531E 03    0.0               0.0               0.0
======================================================================================================

 NODE        TEMP         DT           DDT        GEN RATE       W             H             F          CURE AT 280 F
  -1     0.3991D-07  -0.3393D-09  -0.4062D-10    0.0       0.3991E-31   0.3989E-31   0.4355E-02    0.2492E 00
   1     0.1996D 01  -0.1697D-01  -0.2031D-02    0.0       0.1996E 01  -0.9800E 02  -0.9800E 02    0.3003E 00
   2     0.5982D 01  -0.5078D-01  -0.6078D-02    0.0       0.5982E 01  -0.9402E 02  -0.9401E 02    0.4428E 00
   3     0.9954D 01  -0.8421D-01  -0.1008D-01    0.0       0.9954E 01  -0.9005E 02  -0.9003E 02    0.6339E 00
   4     0.1390D 02  -0.1170D 00  -0.1400D-01    0.0       0.1390E 02  -0.8610E 02  -0.8608E 02    0.8664E 00
   5     0.1781D 02  -0.1489D 00  -0.1782D-01    0.0       0.1781E 02  -0.8219E 02  -0.8216E 02    0.1135E 01
   6     0.2168D 02  -0.1797D 00  -0.2151D-01    0.0       0.2168E 02  -0.7832E 02  -0.7829E 02    0.1434E 01
   7     0.2550D 02  -0.2092D 00  -0.2504D-01    0.0       0.2550E 02  -0.7450E 02  -0.7448E 02    0.1760E 01
   8     0.2925D 02  -0.2371D 00  -0.2838D-01    0.0       0.2925E 02  -0.7075E 02  -0.7072E 02    0.2109E 01
   9     0.3293D 02  -0.2633D 00  -0.3152D-01    0.0       0.3293E 02  -0.6707E 02  -0.6704E 02    0.2476E 01
  10     0.3654D 02  -0.2877D 00  -0.3444D-01    0.0       0.3654E 02  -0.6346E 02  -0.6343E 02    0.2858E 01
  11     0.4006D 02  -0.3100D 00  -0.3711D-01    0.0       0.4006E 02  -0.5994E 02  -0.5991E 02    0.3253E 01
  12     0.4349D 02  -0.3302D 00  -0.3953D-01    0.0       0.4349E 02  -0.5651E 02  -0.5648E 02    0.3656E 01
  13     0.4682D 02  -0.3482D 00  -0.4168D-01    0.0       0.4682E 02  -0.5318E 02  -0.5315E 02    0.4065E 01
  14     0.5005D 02  -0.3638D 00  -0.4355D-01    0.0       0.5005E 02  -0.4995E 02  -0.4993E 02    0.4477E 01
  15     0.5317D 02  -0.3772D 00  -0.4515D-01    0.0       0.5317E 02  -0.4683E 02  -0.4681E 02    0.4889E 01
  16     0.5619D 02  -0.3882D 00  -0.4646D-01    0.0       0.5619E 02  -0.4381E 02  -0.4380E 02    0.5298E 01
  17     0.5908D 02  -0.3968D 00  -0.4750D-01    0.0       0.5908E 02  -0.4092E 02  -0.4091E 02    0.5703E 01
  18     0.6186D 02  -0.4031D 00  -0.4825D-01    0.0       0.6186E 02  -0.3814E 02  -0.3813E 02    0.6100E 01
  19     0.6452D 02  -0.4071D 00  -0.4873D-01    0.0       0.6452E 02  -0.3548E 02  -0.3548E 02    0.6488E 01
  20     0.6706D 02  -0.4089D 00  -0.4895D-01    0.0       0.6706E 02  -0.3294E 02  -0.3294E 02    0.6865E 01
  21     0.7069D 02  -0.4085D 00  -0.4890D-01    0.0       0.1414E 03  -0.5863E 02  -0.5863E 02    0.7421E 01
  22     0.7504D 02  -0.4002D 00  -0.4790D-01    0.0       0.1501E 03  -0.4993E 02  -0.4993E 02    0.8093E 01
  23     0.7891D 02  -0.3851D 00  -0.4610D-01    0.0       0.1578E 03  -0.4218E 02  -0.4218E 02    0.8698E 01
  24     0.8233D 02  -0.3646D 00  -0.4364D-01    0.0       0.1647E 03  -0.3534E 02  -0.3534E 02    0.9234E 01
  25     0.8531D 02  -0.3398D 00  -0.4068D-01    0.0       0.1706E 03  -0.2937E 02  -0.2937E 02    0.9700E 01
  26     0.8789D 02  -0.3121D 00  -0.3735D-01    0.0       0.1758E 03  -0.2421E 02  -0.2421E 02    0.1010E 02
  27     0.9010D 02  -0.2825D 00  -0.3382D-01    0.0       0.1802E 03  -0.1980E 02  -0.1980E 02    0.1043E 02
  28     0.9197D 02  -0.2523D 00  -0.3021D-01    0.0       0.1839E 03  -0.1606E 02  -0.1606E 02    0.1071E 02
  29     0.9354D 02  -0.2224D 00  -0.2662D-01    0.0       0.1871E 03  -0.1292E 02  -0.1292E 02    0.1093E 02
  30     0.9484D 02  -0.1935D 00  -0.2316D-01    0.0       0.1897E 03  -0.1032E 02  -0.1032E 02    0.1111E 02
  31     0.9645D 02  -0.1526D 00  -0.1827D-01    0.0       0.3858E 03  -0.1421E 02  -0.1421E 02    0.1133E 02
  32     0.9785D 02  -0.1073D 00  -0.1284D-01    0.0       0.3914E 03  -0.8585E 01  -0.8586E 01    0.1150E 02
  33     0.9874D 02  -0.7217D-01  -0.8639D-02    0.0       0.3950E 03  -0.5025E 01  -0.5025E 01    0.1160E 02
  34     0.9928D 02  -0.4658D-01  -0.5576D-02    0.0       0.3971E 03  -0.2861E 01  -0.2861E 01    0.1166E 02
  35     0.9960D 02  -0.2926D-01  -0.3503D-02    0.0       0.3984E 03  -0.1602E 01  -0.1602E 01    0.1169E 02
  36     0.9986D 02  -0.1231D-01  -0.1473D-02    0.0       0.7989E 03  -0.1137E 01  -0.1137E 01    0.1171E 02
  37     0.9997D 02  -0.3973D-02  -0.4756D-03    0.0       0.7997E 03  -0.2716E 00  -0.2716E 00    0.1172E 02
  38     0.9999D 02  -0.9891D-03  -0.1184D-03    0.0       0.7999E 03  -0.5662E-01  -0.5662E-01    0.1172E 02
  39     0.1000D 03  -0.2133D-03  -0.2554D-04    0.0       0.8000E 03  -0.1057E-01  -0.1057E-01    0.1172E 02
  40     0.1000D 03  -0.4717D-04  -0.5647D-05    0.0       0.8000E 03  -0.2025E-02  -0.2025E-02    0.1172E 02
======================================================================================================
```

D. CYLINDER PROBLEM

An infinitely long cylinder of unit radius, with a unit surface temperature, is embedded in a medium with a zero initial temperature. TRUMP is used to calculate the temperature distribution in the medium. All thermal properties for the medium have unit values. Results are compared with an analytical solution.

1. Trump Solution

The input data are shown in Figure 8. Because of the geometrical shape of cylinder, the symmetry type indicator, KD, in BLOCK 1 is selected as 2. Maximum problem time, TIMAX, is made 50, and the medium initial temperature, TONE, is set to 0.0. Also, for temperature accuracy, TVARY, is selected as 0.05. In BLOCK 2, unit properties of the medium are defined. Nodal point locations are chosen so as to be able to compare the TRUMP solution with the given table in Reference 5 rather easily. The zoning of the medium is shown in the figure below.



In BLOCK 4, the arithmetic mean average radii, DRAD, are used to obtain correct nodal volumes for cylindrical shaped nodes. Actual node interfaces are used as interface radii, DRAD, in BLOCK 5, to obtain the areas of the connected interfaces. In BLOCK 6, zero-volume surface node, node -1, is connected to the boundary node with a heat transfer coefficient of 1.0×10^8, and geometric factors for external surface node area are defined. Computer running time for this problem was 15.46 sec.

```
*  CYLINDER  PROBLEM.  OCT.  1971.
BLOCK 1   PROBLEM  CONTROLS.
   5         1
   2                          0.05             50.0
   0.0
BLOCK 2  MATERIAL NAMES, NUMBERS, AND THERMAL PROPERTIES.
AAIR    50                    1.0      1.0      1.0

BLOCK 4  NODF NUMBERS, MATERIALS, TYPES, AND DIMENSIONS.
   -1            50                                1.0
    1            50           1.0      0.2      1.1
    2            50           1.0      0.2      1.3
    3            50           1.0      0.2      1.5
    4            50           1.0      0.2      1.7
    5            50           1.0      0.2      1.9
    6            50           1.0      0.4      2.2
    7            50           1.0      0.4      2.6
    8            50           1.0      0.4      3.0
    9            50           1.0      0.4      3.4
   10            50           1.0      0.4      3.8
   11            50           1.0      0.4      4.2
   12            50           1.0      0.4      4.6
   13            50           1.0      0.4      5.0
   14            50           1.0      0.4      5.4
   15            50           1.0      0.4      5.8
   16            50           1.0      0.4      6.2
   17            50           1.0      0.4      6.6
   18            50           1.0      0.4      7.0
   19            50           1.0      0.4      7.4
   20            50           1.0      0.4      7.8
   21            50           1.0      0.4      8.2
   22            50           1.0      0.4      8.6
   23            50           1.0      0.4      9.0
   24            50           1.0      1.6     10.0
   25            50           1.0      1.6     11.6
   26            50           1.0      1.6     13.2
   27            50           1.0      1.6     14.8
   28            50           1.0      1.6     16.4
   29            50           1.0      1.6     18.0
   30            50           1.0      2.4     20.0
   31            50           1.0      2.4     22.4
   32            50           1.0      2.4     24.8
   33            50           1.0      2.4     27.2
   34            50           1.0      3.2     30.0

BLOCK 5  INTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1    1               0.0      0.1      1.0      1.0
    1    2               0.1      0.1      1.0      1.2
```

FIG. 8 INPUT DATA FOR CYLINDER

```
     2    3           0.1       0.1       1.0       1.4
     3    4           0.1       0.1       1.0       1.6
     4    5           0.1       0.1       1.0       1.8
     5    6           0.1       0.2       1.0       2.0
     6    7           0.2       0.2       1.0       2.4
     7    8           0.2       0.2       1.0       2.8
     8    9           0.2       0.2       1.0       3.2
     9   10           0.2       0.2       1.0       3.6
    10   11           0.2       0.2       1.0       4.0
    11   12           0.2       0.2       1.0       4.4
    12   13           0.2       0.2       1.0       4.8
    13   14           0.2       0.2       1.0       5.2
    14   15           0.2       0.2       1.0       5.6
    15   16           0.2       0.2       1.0       6.0
    16   17           0.2       0.2       1.0       6.4
    17   18           0.2       0.2       1.0       6.8
    18   19           0.2       0.2       1.0       7.2
    19   20           0.2       0.2       1.0       7.6
    20   21           0.2       0.2       1.0       8.0
    21   22           0.2       0.2       1.0       8.4
    22   23           0.2       0.2       1.0       8.8
    23   24           0.2       0.8       1.0       9.2
    24   25           0.8       0.8       1.0      10.8
    25   26           0.8       0.8       1.0      12.4
    26   27           0.8       0.8       1.0      14.0
    27   28           0.8       0.8       1.0      15.6
    28   29           0.8       0.8       1.0      17.2
    29   30           0.8       1.2       1.0      18.8
    30   31           1.2       1.2       1.0      21.2
    31   32           1.2       1.2       1.0      23.6
    32   33           1.2       1.2       1.0      26.0
    33   34           1.2       1.6       1.0      28.4
BLOCK 6  EXTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1 2001                      1.0       1.0       1.0     E8

BLOCK 7  BOUNDARY NODE TEMPERATURES.
 2001                 1.0

BLOCK 9  INITIAL CONDITIONS.
   -1                 1.0

ENDED-1  LAST CARD OF DATA DECK.
```

FIG. 8 (CONT'D)

TABLE V

TEMPERATURE DISTRIBUTION FOR CYLINDER (TIME = 50)

R	T (Trump)	T (Analytic)
1.0	1.0	1.0
1.1	0.961	0.963
1.3	0.897	0.898
1.5	0.842	0.843
1.7	0.794	0.794
1.9	0.751	0.751
3.0	0.574	0.574
5.0	0.382	0.381
7.0	0.261	0.260
9.0	0.179	0.177
10.0	0.146	0.146
20.0	0.016	0.016
30.0	0.001	0.001

```
TRUMP OUTPUT DATA                                                                          DATA DECK     1
 *  CYLINDER    PROBLEM
==========================================================================================================
   PRINTOUT     CYCLE    TCC FAST   TCC SLOW     KWIT       DELTMX          SMALL          TVARY       NUTS
      13          53          0          0          1     1.00000E 12    8.53332E-03    5.00000E-02     2

    TOTAL TIME      TIME STEP      HEAT FLOW     TEMP FROM FLUX   FLUX RATE      TEMP RATE
   5.00000E 01    2.46155E-01    1.54873E 02     4.94183E-02    3.09746E 00    9.88366E-04

     AVG TEMP     HEAT CAPACITY   HEAT CONTENT     GEN RATE         HEAT GEN        TEMP FROM GEN
   4.97682E-02    3.13392D 03    1.55970E 02       0.0              0.0              0.0
==========================================================================================================

NODE         TEMP          DT            DDT          GEN RATE          W             H             F          CURE AT 280 F
 -1      0.1000D 01    0.9042D-13    0.1682D-12    0.0          0.1000E-23   -0.1359E-33   -0.8966E-02    0.0
  1      0.9614D 00    0.2641D-04    0.4912D-04    0.0          0.1329E 01    0.1329E 01    0.1315E 01    0.0
  2      0.8971D 00    0.7112D-04    0.1323D-03    0.0          0.1465E 01    0.1465E 01    0.1438E 01    0.0
  3      0.8419D 00    0.1097D-03    0.2041D-03    0.0          0.1587E 01    0.1587E 01    0.1541E 01    0.0
  4      0.7937D 00    0.1449D-03    0.2695D-03    0.0          0.1696E 01    0.1696E 01    0.1633E 01    0.0
  5      0.7509D 00    0.1793D-03    0.3335D-03    0.0          0.1793E 01    0.1793E 01    0.1731E 01    0.0
  6      0.6931D 00    0.2322D-03    0.4318D-03    0.0          0.3832E 01    0.3832E 01    0.3781E 01    0.0
  7      0.6290D 00    0.2859D-03    0.5317D-03    0.0          0.4110E 01    0.4110E 01    0.4057E 01    0.0
  8      0.5742D 00    0.3290D-03    0.6118D-03    0.0          0.4329E 01    0.4329E 01    0.4270E 01    0.0
  9      0.5265D 00    0.3649D-03    0.6786D-03    0.0          0.4499E 01    0.4499E 01    0.4433E 01    0.0
 10      0.4843D 00    0.3954D-03    0.7353D-03    0.0          0.4625E 01    0.4625E 01    0.4555E 01    0.0
 11      0.4466D 00    0.4213D-03    0.7835D-03    0.0          0.4714E 01    0.4714E 01    0.4640E 01    0.0
 12      0.4125D 00    0.4432D-03    0.8243D-03    0.0          0.4769E 01    0.4769E 01    0.4695E 01    0.0
 13      0.3816D 00    0.4617D-03    0.8587D-03    0.0          0.4796E 01    0.4796E 01    0.4724E 01    0.0
 14      0.3534D 00    0.4771D-03    0.8873D-03    0.0          0.4796E 01    0.4796E 01    0.4728E 01    0.0
 15      0.3275D 00    0.4897D-03    0.9108D-03    0.0          0.4774E 01    0.4774E 01    0.4712E 01    0.0
 16      0.3036D 00    0.4998D-03    0.9295D-03    0.0          0.4731E 01    0.4731E 01    0.4677E 01    0.0
 17      0.2816D 00    0.5076D-03    0.9440D-03    0.0          0.4671E 01    0.4671E 01    0.4625E 01    0.0
 18      0.2612D 00    0.5133D-03    0.9547D-03    0.0          0.4595E 01    0.4595E 01    0.4559E 01    0.0
 19      0.2422D 00    0.5172D-03    0.9619D-03    0.0          0.4505E 01    0.4505E 01    0.4478E 01    0.0
 20      0.2246D 00    0.5193D-03    0.9658D-03    0.0          0.4404E 01    0.4404E 01    0.4384E 01    0.0
 21      0.2083D 00    0.5206D-03    0.9683D-03    0.0          0.4292E 01    0.4292E 01    0.4278E 01    0.0
 22      0.1930D 00    0.5243D-03    0.9751D-03    0.0          0.4172E 01    0.4172E 01    0.4162E 01    0.0
 23      0.1788D 00    0.5314D-03    0.9883D-03    0.0          0.4045E 01    0.4045E 01    0.4039E 01    0.0
 24      0.1458D 00    0.5453D-03    0.1014D-02    0.0          0.1465E 02    0.1465E 02    0.1465E 02    0.0
 25      0.1059D 00    0.5021D-03    0.9337D-03    0.0          0.1235E 02    0.1235E 02    0.1235E 02    0.0
 26      0.7611D-01    0.4392D-03    0.8169D-03    0.0          0.1010E 02    0.1010E 02    0.1010E 02    0.0
 27      0.5401D-01    0.3705D-03    0.6891D-03    0.0          0.8036E 01    0.8036E 01    0.8035E 01    0.0
 28      0.3783D-01    0.3030D-03    0.5635D-03    0.0          0.6237E 01    0.6237E 01    0.6236E 01    0.0
 29      0.2616D-01    0.2407D-03    0.4476D-03    0.0          0.4734E 01    0.4734E 01    0.4734E 01    0.0
 30      0.1581D-01    0.1725D-03    0.3208D-03    0.0          0.4769E 01    0.4769E 01    0.4769E 01    0.0
 31      0.8616D-02    0.1110D-03    0.2065D-03    0.0          0.2910E 01    0.2910E 01    0.2910E 01    0.0
 32      0.4621D-02    0.6896D-04    0.1283D-03    0.0          0.1728E 01    0.1728E 01    0.1728E 01    0.0
 33      0.2539D-02    0.4331D-04    0.8055D-04    0.0          0.1041E 01    0.1041E 01    0.1041E 01    0.0
 34      0.1452D-02    0.2811D-04    0.5228D-04    0.0          0.8761E 00    0.8761E 00    0.8760E 00    0.0
----------------------------------------------------------------------------------------------------------
```

E. SPHERE PROBLEM

A sphere with a unit radius and a unit surface temperature is embedded in a medium with a zero initial temperature. The temperature distribution in the medium is calculated using TRUMP assuming all the thermal properties have unit values. Results are compared with the analytical solution.

1. Analytic Solution

$$T(r,t) = \mathrm{erfc}\left(\frac{r-1}{2\ t}\right)$$

2. Trump Solution

The input data from the cylinder problem can be used to solve this problem if the symmetry type indicator, KD, in BLOCK 1 is changed from 2 to 3 to correspond to the centrisymmetric geometry. All other input conditions are identical with the cylinder problem. The computer time used for this problem was 11.61 sec.

TABLE VI

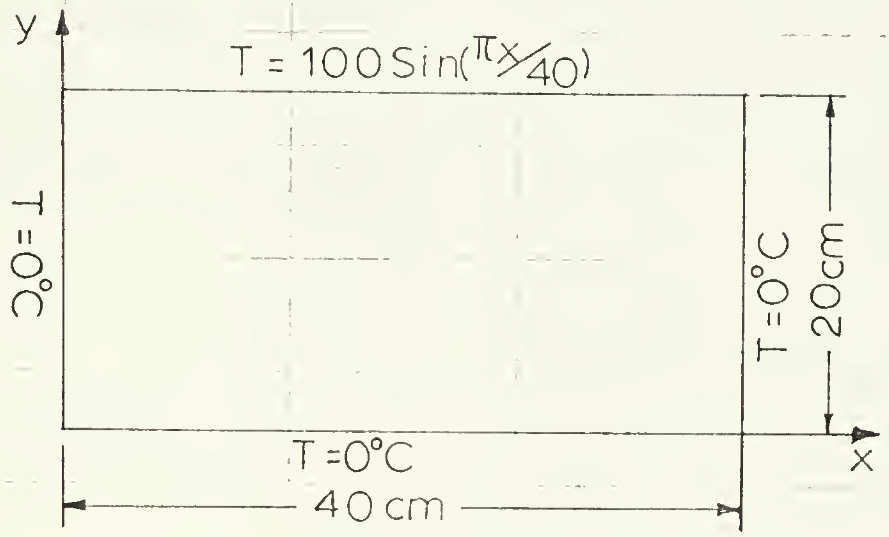
TEMPERATURE DISTRIBUTION FOR SPHERE (TIME = 50)

R	T (Trump)	T (Analytic)
1.0	1.0	1.00
1.1	0.89	0.90
1.3	0.75	0.75
1.5	0.64	0.64
1.7	0.56	0.56
1.9	0.49	0.49
3.0	0.28	0.28
5.0	0.14	0.14
7.0	0.08	0.08
9.0	0.05	0.05
10.0	0.04	0.04
20.0	0.00	0.00
30.0	0.00	0.00

```
TRUMP OUTPUT DATA                                                                                   DATA DECK    1

 *   SPHERE          PROBLEM
==========================================================================================================================

   PRINTOUT     CYCLE    TOO FAST   TOO SLOW     KWIT       DELTMX           SMALL            TVARY        NUTS
     11           43         0          0          1      1.00000E 12     8.49293E-03     5.00000E-02     79

    TOTAL TIME      TIME STEP        HEAT FLOW     TEMP FROM FLUX   FLUX RATE       TEMP RATE
   5.00000E 01     1.43451E 01     7.10860E 02     5.38320E-03     1.42172E 01     1.07664E-04

     AVG TEMP      HEAT CAPACITY     HEAT CONTENT     GEN RATE        HEAT GEN        TEMP FROM GEN
   5.93038E-03     1.32052D 05     7.83115E 02     0.0              0.0             0.0
==========================================================================================================================

  NODE         TEMP           DT              DDT           GE N RATE        W            H             F          CURF AT 280 F
   -1      0.1000D 01     0.2270D-12     0.4634D-14      0.0           0.1000E-23   -0.1307E-34   -0.1158E 00     0.0
    1      0.8946D 00     0.1819D-02     0.3715D-04      0.0           0.2720E 01    0.2720E 01    0.2526E 01     0.0
    2      0.7481D 00     0.4319D-02     0.8819D-04      0.0           0.3177E 01    0.3177E 01    0.2726E 01     0.0
    3      0.6404D 00     0.6111D-02     0.1248D-03      0.0           0.3621E 01    0.3621E 01    0.2787E 01     0.0
    4      0.5579D 00     0.7428D-02     0.1517D-03      0.0           0.4052E 01    0.4052E 01    0.2785E 01     0.0
    5      0.4927D 00     0.8417D-02     0.1718D-03      0.0           0.4470E 01    0.4470E 01    0.3064E 01     0.0
    6      0.4133D 00     0.9563D-02     0.1953D-03      0.0           0.1005E 02    0.1005E 02    0.8643E 01     0.0
    7      0.3398D 00     0.1055D-01     0.2153D-03      0.0           0.1154E 02    0.1154E 02    0.9752E 01     0.0
    8      0.2857D 00     0.1120D-01     0.2287D-03      0.0           0.1293E 02    0.1293E 02    0.1052E 02     0.0
    9      0.2443D 00     0.1166D-01     0.2380D-03      0.0           0.1420E 02    0.1420E 02    0.1116E 02     0.0
   10      0.2116D 00     0.1198D-01     0.2446D-03      0.0           0.1536E 02    0.1536E 02    0.1170E 02     0.0
   11      0.1851D 00     0.1221D-01     0.2494D-03      0.0           0.1641E 02    0.1641E 02    0.1220E 02     0.0
   12      0.1632D 00     0.1238D-01     0.2527D-03      0.0           0.1736E 02    0.1736E 02    0.1269E 02     0.0
   13      0.1448D 00     0.1248D-01     0.2548D-03      0.0           0.1820E 02    0.1820E 02    0.1320E 02     0.0
   14      0.1291D 00     0.1253D-01     0.2558D-03      0.0           0.1893E 02    0.1893E 02    0.1375E 02     0.0
   15      0.1156D 00     0.1252D-01     0.2557D-03      0.0           0.1955E 02    0.1955E 02    0.1434E 02     0.0
   16      0.1039D 00     0.1246D-01     0.2544D-03      0.0           0.2007E 02    0.2007E 02    0.1498E 02     0.0
   17      0.9352D-01     0.1234D-01     0.2519D-03      0.0           0.2048E 02    0.2048E 02    0.1563E 02     0.0
   18      0.8438D-01     0.1217D-01     0.2484D-03      0.0           0.2078E 02    0.2078E 02    0.1528E 02     0.0
   19      0.7624D-01     0.1194D-01     0.2438D-03      0.0           0.2099E 02    0.2099E 02    0.1687E 02     0.0
   20      0.6896D-01     0.1167D-01     0.2382D-03      0.0           0.2109E 02    0.2109E 02    0.1739E 02     0.0
   21      0.6241D-01     0.1135D-01     0.2318D-03      0.0           0.2109E 02    0.2109E 02    0.1779E 02     0.0
   22      0.5650D-01     0.1101D-01     0.2247D-03      0.0           0.2100E 02    0.2100E 02    0.1803E 02     0.0
   23      0.5115D-01     0.1064D-01     0.2172D-03      0.0           0.2083E 02    0.2083E 02    0.1895E 02     0.0
   24      0.3906D-01     0.9648D-02     0.1970D-03      0.0           0.7854E 02    0.7854E 02    0.7784E 02     0.0
   25      0.2628D-01     0.7858D-02     0.1604D-03      0.0           0.7110E 02    0.7110E 02    0.7072E 02     0.0
   26      0.1770D-01     0.6187D-02     0.1263D-03      0.0           0.6201E 02    0.6201E 02    0.6179E 02     0.0
   27      0.1190D-01     0.4741D-02     0.9680D-04      0.0           0.5240E 02    0.5240E 02    0.5232E 02     0.0
   28      0.7973D-02     0.3547D-02     0.7242D-04      0.0           0.4312E 02    0.4312E 02    0.4313E 02     0.0
   29      0.5326D-02     0.2597D-02     0.5302D-04      0.0           0.3469E 02    0.3469E 02    0.3476E 02     0.0
   30      0.3108D-02     0.1683D-02     0.3436D-04      0.0           0.3749E 02    0.3749E 02    0.3756E 02     0.0
   31      0.1664D-02     0.9838D-03     0.2021D-04      0.0           0.2518E 02    0.2518E 02    0.2525E 02     0.0
   32      0.8999D-03     0.5787D-03     0.1182D-04      0.0           0.1669E 02    0.1669E 02    0.1575E 02     0.0
   33      0.5146D-03     0.3531D-03     0.7209D-05      0.0           0.1148E 02    0.1148E 02    0.1153E 02     0.0
   34      0.3181D-03     0.2308D-03     0.4712D-05      0.0           0.1151E 02    0.1151E 02    0.1153E 02     0.0
==========================================================================================================================
```

F. TWO DIMENSIONAL PLATE PROBLEM

A rectangular plate, 40 x 20 cm, has a thermal conductivity of 400 cal/cm-sec-°C, a density of 1.0 g/cm^3, and a heat capacity of 1.0 cal/g-°C and is initially at 0.0°C. Three sides of the plate are maintained at 0.0°C. The upper side of the plate has a temperature distribution that varies sinusoidally with x, starting at 0°C, reaching a maximum of 100°C at the midpoint and decreasing to 0°C at the outer edge (see figure below). The steady-state temperature distribution is calculated in the plate using TRUMP. Results are compared with the analytical solution.



1. Analytic Solution

$$T(x,y) = 100 \frac{\sinh(\frac{\pi y}{40})}{\sinh(\frac{\pi}{2})} \sin(\frac{\pi x}{40})$$

where $0 \leqslant x \leqslant 40$ and $0 \leqslant y \leqslant 20$

2. Trump Solution

The input data are shown in Figure 9. Because the boundary conditions of the plate and the plate itself are symmetric with respect to the plane at x = 20 cm, this symmetry plane is assumed to be an insulated surface. The zoning of the plate is shown in Figure 10. In BLOCK 1 of the input data, IPRINT, is selected as 9999, so that printouts are obtained only for the first, second, and last time steps. To obtain a rapid conclusion to the transient problem, TVARY, is selected as $100.0^{\circ}C$. In BLOCK 4, instead of using a very large number of surface nodes for sides x = 40 cm and y = 0, surface node -1 and surface node -4 are used, respectively, in order to reduce the number of the input cards, and in BLOCK 6 they are connected to the boundary nodes 2001 and 2004. In BLOCK 7, the boundary node temperatures, and in BLOCK 9, the initial temperatures of the surface nodes are given. The computer running time for this problem was 28.71 sec.

```
*  TWO-DIMENSIONAL SLAB PROBLEM.  NOVEMBER 1971
BLOCK 1  PROBLEM LIMITS AND CONTROLS.
 9999          1
    1                             100.0
  0.0
BLOCK 2  MATERIAL NAME AND NUMBER, THERMAL PROPERTIES.
AAG      50                        1.0        1.0        400.0

BLOCK 4   NODE NUMBERS, MATERIALS, TYPES, AND DIMENSIONS.
    1    99    1    50             2.0        2.0        1.0
   -1               50
   -4               50
  -31               50
  -32               50
  -33               50
  -34               50
  -35               50
  -36               50
  -37               50
  -38               50
  -39               50
  -40               50

BLOCK 5  INTERNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
    1    2   8   1   1   1.0       1.0        2.0        1.0
   11   12   8   1   1   1.0       1.0        2.0        1.0
   21   22   8   1   1   1.0       1.0        2.0        1.0
   31   32   8   1   1   1.0       1.0        2.0        1.0
   41   42   8   1   1   1.0       1.0        2.0        1.0
   51   52   8   1   1   1.0       1.0        2.0        1.0
   61   62   8   1   1   1.0       1.0        2.0        1.0
   71   72   8   1   1   1.0       1.0        2.0        1.0
   81   82   8   1   1   1.0       1.0        2.0        1.0
   91   92   8   1   1   1.0       1.0        2.0        1.0
    1   11  89   1   1   1.0       1.0        2.0        1.0
   -1    1   9       1             1.0        2.0        1.0
   -4    1   9      10             1.0        2.0        1.0
  -31   91                         1.0        2.0        1.0
  -32   92                         1.0        2.0        1.0
  -33   93                         1.0        2.0        1.0
  -34   94                         1.0        2.0        1.0
  -35   95                         1.0        2.0        1.0
  -36   96                         1.0        2.0        1.0
  -37   97                         1.0        2.0        1.0
  -38   98                         1.0        2.0        1.0
  -39   99                         1.0        2.0        1.0
  -40  100                         1.0        2.0        1.0
```

FIG. 9 INPUT DATA FOR TWO-DIMENSIONAL-PLATE

```
BLOCK 6  EXTEPNAL CONNECTION NODE NUMBERS AND DESCRIPTIONS.
   -1 2001                        20.0      1.0       1.0      E8
   -4 2004                        20.0      1.0       1.0      E8
  -31  310                        2.0       1.0       1.0      E8
  -32  320                        2.0       1.0       1.0      E8
  -33  330                        2.0       1.0       1.0      E8
  -34  340                        2.0       1.0       1.0      E8
  -35  350                        2.0       1.0       1.0      E8
  -36  360                        2.0       1.0       1.0      E8
  -37  370                        2.0       1.0       1.0      E8
  -38  380                        2.0       1.0       1.0      E8
  -39  390                        2.0       1.0       1.0      E8
  -40  400                        2.0       1.0       1.0      E8

BLOCK 7  BOUNDARY NODE TEMPERATURES.
 2001            0.0
 2004            0.0
  310            7.84590964
  320            23.3445364
  330            38.2683433
  340            52.2498565
  350            64.9448049
  360            76.0405968
  370            85.2640186
  380            92.3879660
  390            97.2370492
  400            99.6919339

BLOCK 9  INITIAL CONDITIONS.
   -1             0.0
   -4             0.0

ENDED-1  LAST CARD OF DATA DECK.
```

FIG. 9 (CONT'D)

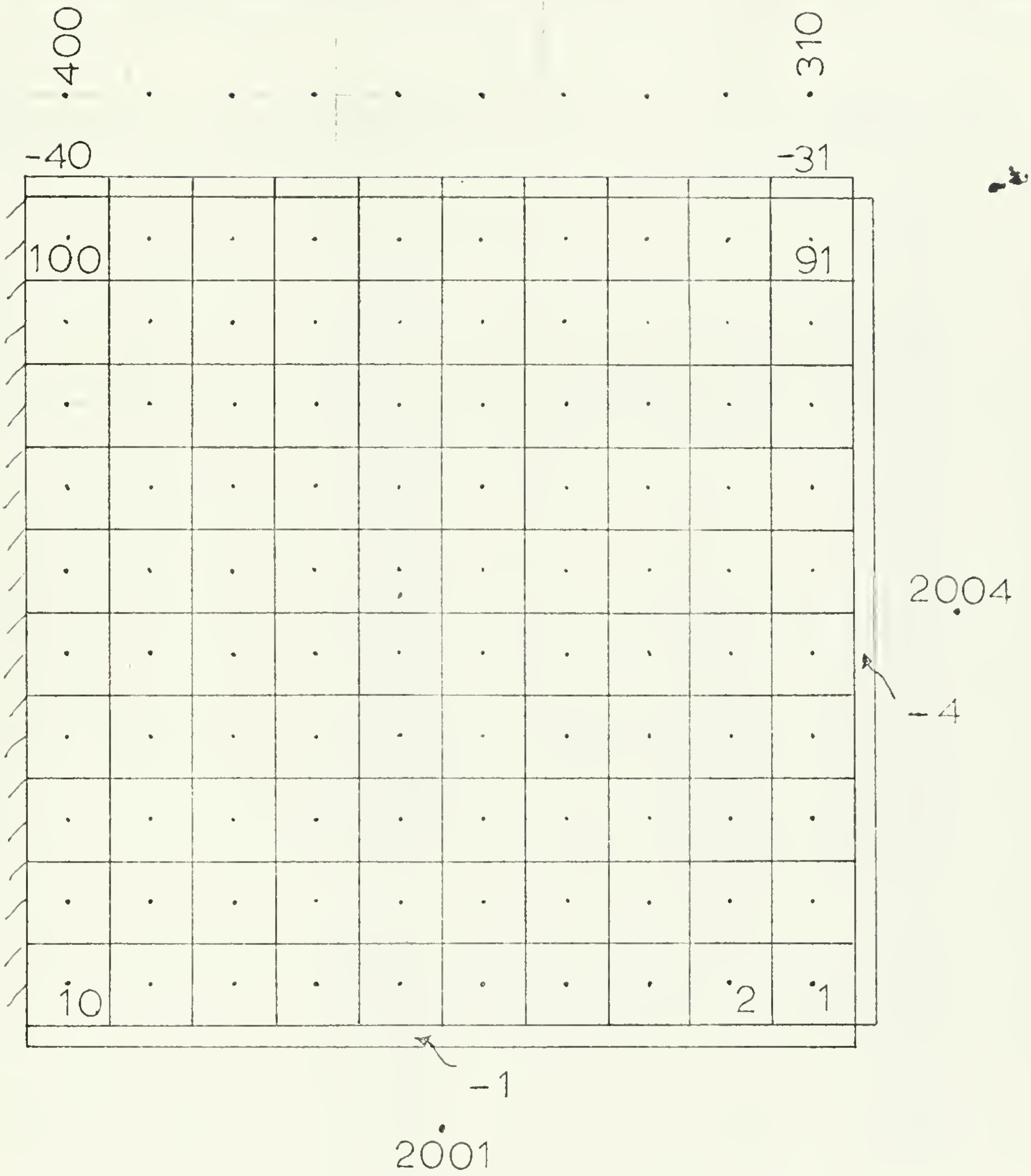


FIG. 10 ZONING OF THE TWO DIMENSIONAL PLATE PROBLEM

TABLE VII

TEMPERATURE DISTRIBUTION FOR TWO-DIMENSIONAL PLATE IN STEADY-STATE

Node Number	T (Trump)	T (Analytic)
1	0.26	0.27
5	2.17	2.23
10	3.34	3.41
11	0.79	0.81
15	6.57	6.71
20	10.10	10.30
21	1.35	1.37
25	11.14	11.37
30	17.13	17.45
31	1.93	1.97
35	16.02	16.31
40	24.63	25.03
41	2.58	2.62
45	21.32	21.65
50	32.77	33.24
51	3.28	3.33
55	27.19	27.53
60	41.77	42.26
61	4.08	4.12
65	33.75	34.09
70	51.84	52.33
71	4.97	5.01
75	41.18	41.49
80	63.24	63.69
81	6.00	6.03
85	49.65	49.92
90	76.23	76.62
91	7.17	7.20
95	59.36	59.58
100	91.12	91.45

```
TRUMP OUTPUT DATA                                                                                   DATA DECK    1
  *  TWO-DIMENSIONAL SLAB PROBLEM.  NOVEMBER 1971
====================================================================================================================
      PRINTOUT     CYCLE    TCC FAST   TCC SLOW      KWIT       DELTMX            SMALL           TVARY        NLTS
         3           54         0          3           4      1.00000E 12      1.66667E-05      1.00000E 02     0

       TOTAL TIME       TIME STEP        HEAT FLOW       TEMP FROM FLUX    FLUX RATE        TEMP RATE
     3.25100E 12      1.00000E 12      2.37981E 15      5.94952E 12      7.32024E 02      1.83006E 00

       AVG TEMP       HEAT CAPACITY    HEAT CONTENT       GEN RATE         HEAT GEN        TEMP FROM GEN
     2.63375E 01      4.00000D 02      1.05350E 04      0.0              0.0              0.0
====================================================================================================================

   NODE         TEMP           DT            DDT         GEN RATE        W             H             F          CURE AT 280 F
     1      0.2619D 00    0.1587D-03    0.1587D-39    0.0         0.1047E 01    0.1047E 01    0.1169E 13    0.0
     2      0.7793D 00    0.4663D-03    0.4663D-39    0.0         0.3117E 01    0.3117E 01    0.2872E 13    0.0
     3      0.1278D 01    0.7596D-03    0.7596D-39    0.0         0.5111E 01    0.5111E 01    0.4624E 13    0.0
     4      0.1745D 01    0.1030D-02    0.1030D-38    0.0         0.6979E 01    0.6979E 01    0.6306E 13    0.0
     5      0.2169D 01    0.1267D-02    0.1267D-38    0.0         0.8676E 01    0.8676E 01    0.7660E 13    0.0
     6      0.2540D 01    0.1465D-02    0.1465D-38    0.0         0.1016E 02    0.1016E 02    0.8968E 13    0.0
     7      0.2849D 01    0.1615D-02    0.1615D-38    0.0         0.1140E 02    0.1140E 02    0.9852E 13    0.0
     8      0.3089D 01    0.1717D-02    0.1717D-38    0.0         0.1235E 02    0.1235E 02    0.1056E 14    0.0
     9      0.3252D 01    0.1765D-02    0.1765D-38    0.0         0.1301E 02    0.1301E 02    0.1086E 14    0.0
    10      0.3336D 01    0.1762D-02    0.1762D-38    0.0         0.1334E 02    0.1334E 02    0.8676E 13    0.0
    11      0.7929D 00    0.4554D-03    0.4554D-39    0.0         0.3172E 01    0.3171E 01    0.2807E 13    0.0
    12      0.2360D 01    0.1338D-02    0.1338D-38    0.0         0.9439E 01    0.9439E 01    0.6540E 13    0.0
    13      0.3869D 01    0.2180D-02    0.2180D-38    0.0         0.1547E 02    0.1547E 02    0.1071E 14    0.0
    14      0.5283D 01    0.2954D-02    0.2954D-38    0.0         0.2113E 02    0.2113E 02    0.1429E 14    0.0
    15      0.6568D 01    0.3637D-02    0.3637D-38    0.0         0.2627E 02    0.2627E 02    0.1780E 14    0.0
    16      0.7692D 01    0.4203D-02    0.4203D-38    0.0         0.3077E 02    0.3077E 02    0.2033E 14    0.0
    17      0.8627D 01    0.4639D-02    0.4639D-38    0.0         0.3451E 02    0.3451E 02    0.2274E 14    0.0
    18      0.9351D 01    0.4929D-02    0.4929D-38    0.0         0.3741E 02    0.3740E 02    0.2411E 14    0.0
    19      0.9846D 01    0.5071D-02    0.5071D-38    0.0         0.3938E 02    0.3938E 02    0.2493E 14    0.0
    20      0.1010D 02    0.5061D-02    0.5061D-38    0.0         0.4040E 02    0.4040E 02    0.1866E 14    0.0
    21      0.1345D 01    0.7069D-03    0.7069D-39    0.0         0.5382E 01    0.5382E 01    0.4332E 13    0.0
    22      0.4004D 01    0.2078D-02    0.2078D-38    0.0         0.1602E 02    0.1602E 02    0.1024E 14    0.0
    23      0.6565D 01    0.3384D-02    0.3384D-38    0.0         0.2626E 02    0.2626E 02    0.1645E 14    0.0
    24      0.8964D 01    0.4589D-02    0.4589D-38    0.0         0.3586E 02    0.3586E 02    0.2251E 14    0.0
    25      0.1114D 02    0.5649D-02    0.5649D-38    0.0         0.4458E 02    0.4458E 02    0.2721E 14    0.0
    26      0.1305D 02    0.6534D-02    0.6534D-38    0.0         0.5220E 02    0.5220E 02    0.3195E 14    0.0
    27      0.1464D 02    0.7210D-02    0.7210D-38    0.0         0.5855E 02    0.5855E 02    0.3490E 14    0.0
    28      0.1587D 02    0.7667D-02    0.7667D-38    0.0         0.6346E 02    0.6346E 02    0.3755E 14    0.0
    29      0.1671D 02    0.7887D-02    0.7887D-38    0.0         0.6682E 02    0.6682E 02    0.3855E 14    0.0
    30      0.1713D 02    0.7874D-02    0.7874D-38    0.0         0.6854E 02    0.6854E 02    0.2893E 14    0.0
    31      0.1934D 01    0.8892D-03    0.8892D-39    0.0         0.7737E 01    0.7737E 01    0.5481E 13    0.0
    32      0.5756D 01    0.2612D-02    0.2612D-38    0.0         0.2303E 02    0.2303E 02    0.1276E 14    0.0
    33      0.9437D 01    0.4258D-02    0.4258D-38    0.0         0.3775E 02    0.3775E 02    0.2097E 14    0.0
    34      0.1289D 02    0.5771D-02    0.5771D-38    0.0         0.5155E 02    0.5155E 02    0.2789E 14    0.0
    35      0.1602D 02    0.7110D-02    0.7110D-38    0.0         0.6408E 02    0.6408E 02    0.3483E 14    0.0
    36      0.1876D 02    0.8223D-02    0.8223D-38    0.0         0.7504E 02    0.7504E 02    0.3950E 14    0.0
    37      0.2104D 02    0.9083D-02    0.9083D-38    0.0         0.8417E 02    0.8417E 02    0.4435E 14    0.0
    38      0.2281D 02    0.9658D-02    0.9658D-38    0.0         0.9122E 02    0.9122E 02    0.4673E 14    0.0
    39      0.2401D 02    0.9942D-02    0.9942D-38    0.0         0.9604E 02    0.9604E 02    0.4851E 14    0.0
    40      0.2463D 02    0.9922D-02    0.9922D-38    0.0         0.9851E 02    0.9851E 02    0.3621E 14    0.0
    41      0.2575D 01    0.9836D-03    0.9836D-39    0.0         0.1030E 02    0.1030E 02    0.6023E 13    0.0
    42      0.7662D 01    0.2891D-02    0.2891D-38    0.0         0.3065E 02    0.3065E 02    0.1425E 14    0.0
    43      0.1256D 02    0.4711D-02    0.4711D-38    0.0         0.5025E 02    0.5025E 02    0.2293E 14    0.0
    44      0.1715D 02    0.6390D-02    0.6390D-38    0.0         0.6861E 02    0.6861E 02    0.3143E 14    0.0
```

```
NODE      TEMP          DT            DDT         GEN RATE       h            H             F          CURE AT 280 F
 45    0.2132D 02   0.7871D-02   0.7871D-38    0.0          0.8529E 02   0.8529E 02   0.3786E 14    0.0
 46    0.2497D 02   0.9112D-02   0.9112D-38    0.0          0.9988E 02   0.9988E 02   0.4454E 14    0.0
 47    0.2800D 02   0.1007D-01   0.1007D-37    0.0          0.1120E 03   0.1120E 03   0.4827E 14    0.0
 48    0.3035D 02   0.1071D-01   0.1071D-37    0.0          0.1214E 03   0.1214E 03   0.5212E 14    0.0
 49    0.3135D 02   0.1103D-01   0.1103D-37    0.0          0.1278E 03   0.1278E 03   0.5319E 14    0.0
 50    0.3277D 02   0.1101D-01   0.1101D-37    0.0          0.1311E 03   0.1311E 03   0.4011E 14    0.0
 51    0.3283D 01   0.9822D-03   0.9822D-39    0.0          0.1313E 02   0.1313E 02   0.6038E 13    0.0
 52    0.9769D 01   0.2886D-02   0.2886D-38    0.0          0.3908E 02   0.3908E 02   0.1412E 14    0.0
 53    0.1602D 02   0.4705D-02   0.4705D-38    0.0          0.6406E 02   0.6406E 02   0.2318E 14    0.0
 54    0.2187D 02   0.6380D-02   0.6380D-38    0.0          0.8748E 02   0.8748E 02   0.3095E 14    0.0
 55    0.2719D 02   0.7865D-02   0.7865D-38    0.0          0.1087E 03   0.1087E 03   0.3861E 14    0.0
 56    0.3183D 02   0.9104D-02   0.9104D-38    0.0          0.1273E 03   0.1273E 03   0.4366E 14    0.0
 57    0.3570D 02   0.1007D-01   0.1007D-37    0.0          0.1428E 03   0.1428E 03   0.4906E 14    0.0
 58    0.3869D 02   0.1071D-01   0.1071D-37    0.0          0.1548E 03   0.1548E 03   0.5127E 14    0.0
 59    0.4073D 02   0.1104D-01   0.1104D-37    0.0          0.1629E 03   0.1629E 03   0.5349E 14    0.0
 60    0.4177D 02   0.1102D-01   0.1102D-37    0.0          0.1671E 03   0.1671E 03   0.3964E 14    0.0
 61    0.4076D 01   0.8847D-03   0.8847D-39    0.0          0.1631E 02   0.1631E 02   0.5420E 13    0.0
 62    0.1213D 02   0.2601D-02   0.2601D-38    0.0          0.4852E 02   0.4852E 02   0.1278E 14    0.0
 63    0.1989D 02   0.4238D-02   0.4238D-38    0.0          0.7954E 02   0.7954E 02   0.2071E 14    0.0
 64    0.2715D 02   0.5751D-02   0.5751D-38    0.0          0.1086E 03   0.1086E 03   0.2829E 14    0.0
 65    0.3375D 02   0.7087D-02   0.7087D-38    0.0          0.1350E 03   0.1350E 03   0.3428E 14    0.0
 66    0.3952D 02   0.8212D-02   0.8212D-38    0.0          0.1581E 03   0.1581E 03   0.4020E 14    0.0
 67    0.4432D 02   0.9080D-02   0.9080D-38    0.0          0.1773E 03   0.1773E 03   0.4343E 14    0.0
 68    0.4803D 02   0.9673D-02   0.9673D-38    0.0          0.1921E 03   0.1921E 03   0.4697E 14    0.0
 69    0.5056D 02   0.9963D-02   0.9963D-38    0.0          0.2022E 03   0.2022E 03   0.4749E 14    0.0
 70    0.5184D 02   0.9951D-02   0.9951D-38    0.0          0.2074E 03   0.2074E 03   0.3608E 14    0.0
 71    0.4974D 01   0.7020D-03   0.7020D-39    0.0          0.1990E 02   0.1990E 02   0.4300E 13    0.0
 72    0.1480D 02   0.2062D-02   0.2062D-38    0.0          0.5920E 02   0.5920E 02   0.1010E 14    0.0
 73    0.2426D 02   0.3363D-02   0.3363D-38    0.0          0.9706E 02   0.9706E 02   0.1650E 14    0.0
 74    0.3313D 02   0.4561D-02   0.4561D-38    0.0          0.1325E 03   0.1325E 03   0.2224E 14    0.0
 75    0.4118D 02   0.5624D-02   0.5624D-38    0.0          0.1647E 03   0.1647E 03   0.2761E 14    0.0
 76    0.4822D 02   0.6515D-02   0.6515D-38    0.0          0.1929E 03   0.1929E 03   0.3141E 14    0.0
 77    0.5407D 02   0.7211D-02   0.7211D-38    0.0          0.2163E 03   0.2163E 03   0.3521E 14    0.0
 78    0.5860D 02   0.7681D-02   0.7681D-38    0.0          0.2344E 03   0.2344E 03   0.3660E 14    0.0
 79    0.6168D 02   0.7918D-02   0.7918D-38    0.0          0.2467E 03   0.2467E 03   0.3835E 14    0.0
 80    0.6324D 02   0.7905D-02   0.7905D-38    0.0          0.2530E 03   0.2530E 03   0.2816E 14    0.0
 81    0.5998D 01   0.4511D-03   0.4511D-39    0.0          0.2399E 02   0.2399E 02   0.2766E 13    0.0
 82    0.1785D 02   0.1326D-02   0.1326D-38    0.0          0.7138E 02   0.7138E 02   0.6491E 13    0.0
 83    0.2925D 02   0.2161D-02   0.2161D-38    0.0          0.1170E 03   0.1170E 03   0.1057E 14    0.0
 84    0.3994D 02   0.2932D-02   0.2932D-38    0.0          0.1598E 03   0.1598E 03   0.1437E 14    0.0
 85    0.4965D 02   0.3615D-02   0.3615D-38    0.0          0.1986E 03   0.1986E 03   0.1758E 14    0.0
 86    0.5813D 02   0.4190D-02   0.4190D-38    0.0          0.2325E 03   0.2325E 03   0.2054E 14    0.0
 87    0.6519D 02   0.4636D-02   0.4636D-38    0.0          0.2608E 03   0.2607E 03   0.2226E 14    0.0
 88    0.7064D 02   0.4943D-02   0.4943D-38    0.0          0.2826E 03   0.2825E 03   0.2408E 14    0.0
 89    0.7435D 02   0.5094D-02   0.5094D-38    0.0          0.2974E 03   0.2974E 03   0.2417E 14    0.0
 90    0.7623D 02   0.5089D-02   0.5089D-38    0.0          0.3049E 03   0.3049E 03   0.1848E 14    0.0
 91    0.7171D 01   0.1571D-03   0.1571D-39    0.0          0.2868E 02   0.2868E 02   0.1156E 13    0.0
 92    0.2134D 02   0.4615D-03   0.4615D-39    0.0          0.8534E 02   0.8534E 02   0.2827E 13    0.0
 93    0.3498D 02   0.7523D-03   0.7523D-39    0.0          0.1399E 03   0.1399E 03   0.4595E 13    0.0
 94    0.4775D 02   0.1020D-02   0.1020D-38    0.0          0.1910E 03   0.1910E 03   0.6226E 13    0.0
 95    0.5936D 02   0.1259D-02   0.1259D-38    0.0          0.2374E 03   0.2374E 03   0.7707E 13    0.0
 96    0.6950D 02   0.1459D-02   0.1459D-38    0.0          0.2780E 03   0.2780E 03   0.8837E 13    0.0
 97    0.7793D 02   0.1615D-02   0.1615D-38    0.0          0.3117E 03   0.3117E 03   0.9891E 13    0.0
 98    0.8444D 02   0.1721D-02   0.1721D-38    0.0          0.3378E 03   0.3378E 03   0.1027E 14    0.0
 99    0.8887D 02   0.1775D-02   0.1775D-38    0.0          0.3555E 03   0.3555E 03   0.1078E 14    0.0
100    0.9112D 02   0.1773D-02   0.1773D-38    0.0          0.3645E 03   0.3645E 03   0.8389E 13    0.0
 -1    0.8514D-05   0.4960D-08   0.4960D-44    0.0          0.8514E-29   0.8514E-29   0.3042E 14    0.0
 -4    0.1296D-04   0.2632D-08   0.2632D-44    0.0          0.1296E-28   0.1296E-28   0.1616E 14    0.0
```

```
NODE       TEMP         DT           DDT         GEN RATE      W            H            F          CURE AT 280 F
 -31     0.7846D 01   0.6486D-09   0.6486D-45   0.0         0.7846E-23   0.7846E-23   0.3986E 12   0.0
 -32     0.2334D 02   0.1906D-08   0.1906D-44   0.0         0.2334E-22   0.2334E-22   0.1169E 13   0.0
 -33     0.3827D 02   0.3106D-08   0.3106D-44   0.0         0.3827E-22   0.3827E-22   0.1900E 13   0.0
 -34     0.5225D 02   0.4215D-08   0.4215D-44   0.0         0.5225E-22   0.5225E-22   0.2568E 13   0.0
 -35     0.6494D 02   0.5196D-08   0.5196D-44   0.0         0.6494E-22   0.6494E-22   0.3164E 13   0.0
 -36     0.7604D 02   0.6026D-08   0.6026D-44   0.0         0.7604E-22   0.7604E-22   0.3696E 13   0.0
 -37     0.8526D 02   0.6670D-08   0.6670D-44   0.0         0.8526E-22   0.8526E-22   0.4026E 13   0.0
 -38     0.9239D 02   0.7114D-08   0.7114D-44   0.0         0.9239E-22   0.9239E-22   0.4352E 13   0.0
 -39     0.9724D 02   0.7334D-08   0.7334D-44   0.0         0.9724E-22   0.9724E-22   0.4345E 13   0.0
 -40     0.9969D 02   0.7329D-08   0.7329D-44   0.0         0.9969E-22   0.9969E-22   0.4441E 13   0.0
=================================================================================================================
```

IV. CONCLUSIONS

The adaptation of the computer program "TRUMP" to the Naval Postgraduate School IBM/360 Model 67 has been achieved, with success. Analytical temperature distributions of various transient and steady-state heat conduction problems compared very well with the results obtained from TRUMP. The complete adaptation has not yet been established, however. At present, punched output and plotted graphs have not been obtained. The computer solutions of the example problems did not require the use of subroutines CHEM, GEN, and FLOW. Thus, the adaptation of these subroutines is yet to be evaluated. Since, the accuracy of the solutions are dependent on the zoning, this effort can be reduced if an existing program "FED: A computer program to generate geometric input for the heat-transfer code TRUMP" by Dale A. Schauer [Reference 4] is adapted to the IBM/360 Model 67 version of TRUMP.

The example problems presented herein will provide a useful reference for preparation of input data for future problems.

TRUMP LISTING

```
C**** LISTING 16050006 *****
C     FORTRAN             HEART   TRUMP LOGIC CONTROL SUB.  FORTRAN MON-40
C""   VERSION 02/20/68  17.3 FOR CDC-6600 ONLY.  (*XEQ OR *CONTROLLEE)
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT
      PEAL*8 FOR,SLIM,T,GEOM
      REAL*8 DFS
      REAL*8 TBS
      REAL*8 ERROR,ERRORX
      REAL*8 HMELT,A
      REAL*8 DELF1,DELF2
      REAL*8 NAME,CAPS
      REAL*8 FLOWN
      REAL*8 FB
      DIMENSION ABLOCK(12)
      REAL*8 ADATA,ABLOCK
      REAL*8 CON,DA,ZIP,TRAN
      REAL*8 AREA,DEL1,DEL2
      REAL*8 AREAS,HSURE
      REAL*8 TRANS
      COMMON /CURE/ EQCURE(355)
      COMMON /AFINK/ AREA(950),DEL1(950),DEL2(950),HINT(950),
     1       NOX1(950),NOX2(950),RINT(950)
      COMMON /ACHEMS/ KA(15),KAX(15),KB(15),KBX(15),
     1       B(355),BB(355),DB(355),DDB(355)
      COMMON /AFLOW/ DELF1(50),DELF2(50),FLAPS(50),FLOWT(12,50),
     1       FLINT(355),FLIPS(355),FLOPS(355),FLOUT(355),
     2       LTABFL(50),NOXF1(50),NOXF2(50),
     3       SLOFL(12,50),TVARFL(12,50)
      COMMON /APLOT/ NODEP(12),NOXEP(12)
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NP,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODES,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NOPADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
```

```
     3        VOL(355),W(355),
     4        DFI(950),FI(950),NOD1(950),NOD2(950),
     5        F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1        LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2        TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4        NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1        NOXE(355),NTYPES(355),TT(355)
      COMMON /AFLOWS/ FLOWN(50),NODF1(50),NODF2(50)
      COMMON /ASPECK/ ERROR(355),ERRORX(355)
      COMMON /ASURE/   AREAS(60),HSURF(60),FB(20),TBS(20),
     1        FS(60),HSURT(12,60),LTABH(60),NOXS(60),NODSB(60),
     2        NOXSB(60),POWER(60),PSURE(60),SLOH(12,60),TVARH(12,60),
     3        LTABT(20),NODB(20),SLOT(12,20),TB(20),TEMPB(12,20),
     4        TIMEB(12,20)
      COMMON /ASURES/ DFS(60),TRANS(60),NODS(60)
      COMMON /AGEN/ GT(12,5),LTABG(5),NODG(5),NOXG(5),
     1        SLOG(12,5),TVARG(12,5)
      COMMON /ACHEM/ AKEM(5),ET(12,5),KEM(5),LTABE(5),LTABQ(5),
     1        LTABZ(5),QT(12,5),SLOE(12,5),SLOQ(12,5),
     2        SLOZ(12,5),TVARE(12,5),TVARQ(12,5),
     3        TVARZ(12,5),ZT(12,5),
     4        EA(355),EB(355),QA(355),QB(355),ZA(355),ZB(355)
  806 FORMAT(10X,'KWIT...1=TIMAX, 2=TMAX, 3=TMIN, 4=S.S., 5=DATA, 6=CARD     0720
     1 OR  KEY, 7=MCYC, 8=MSEC, 9=BLK 2 OR 4'/, 15X,'10=CONVERGENCE FAIL     0730
     2URE, 11=PROBLEM LIMIT, 12= TABLE LENGTH')                               0740
  825 FORMAT(10X,14HMACHINE TIME (,A6,4H) = ,I5,2X,7HSECONDS)                0750
  830 FORMAT(38X,39HTHIS DATA ADDS TO DATA ALREADY READ IN.)                 0760
  840 FORMAT(A5,I2,I1,12A6)                                                   0770
  845 FORMAT(10X,4HDATA,1X,A5,I3,I1,14X,12A6)                                 0780
  850 FORMAT(38X,12A6)                                                        0790
  855 FORMAT(A1,14A5,A1,4A2)                                                  0800
  865 FORMAT(10X,100(1H=))                                                    0810
  905 FORMAT(10X,12HINPUT UNIT =,I4,16H.  OUTPUT UNIT =,I5,1H.)              0820
  945 FORMAT(10X,17HEND OF TRUMP JOB.)                                        0830
C$$$$$**************THIS IS INITIAL ENTRY POINT FOR /* XEQ/ JOBS.********    0840
      M1=1                                                                    0850
      M2=15                                                                   0860
      M3=5                                                                    0870
      M4=355                                                                  0880
      M5=950                                                                  0890
      M6=60                                                                   0900
      M7=20                                                                   0910
      M8=5                                                                    0920
      M9=12                                                                   0930
      M10=50                                                                  0940
```

```
      M11=10
      M12=75
      DO 5 N=1,355
    5 EQCURE(N)=0.0
      NVARFL=0
      NOFLOW=0
      NVARO=0
      NVARZ=0
      NOGEN=0
      NKEM=0
      NOCON=0
      NOSCON=0
      NMELT=0
      NOSPEC=0
      NVARE=0
      NVARG=0
      NTABS=0
      NRS=5
      MW=6
  805 FORMAT(10X,10HENDED PROB,I5,5X,7HKCYC = ,I5,5X,7HKWIT = ,I2,5X,9HS
     1 UMTIM = ,E12.5)
   20 NPROB = 0
      NR = NRS
      M = MW
      NDATA=0
      NSAVE=0
   40 NPROB = NPROB + 1
      KCYC = -1
CCC   INITIALIZE CONTROL PARAMETERS.
      NCHECK=0
      NOW=0
      NOFF=0
      NDUMP=0
      NDISC=0
      NSTOP=0
      KWIT=0
      IBLOCK=0
      DO 45 N = 1,15
      NEWBL(N) = 0
   45 CONTINUE
      DO 48 N = 1,20
      NX1(N) =0
   48 CONTINUE
   50 READ(5,855)NAME
      WRITE(6,855)NAME
C--------COME HERE WHEN PROBLEM I. D. HAS BEEN READ IN.
   72 WRITE(6,905)NR,M
CCC   ZERO ALL COUNTERS WHEN NO DATA IS CARRIED OVER FROM LAST PROBLEM.
```

```
      NOSPEC=0
CCC   SAVE INITIAL NUMBER OF ITEMS IN EACH BLOCK.
  100 DO 102 N = 1,12
      ITEMS(N) = NOSPEC
  102 CONTINUE
C---------READ IN BLOCK NUMBER CARDS.
  105 WRITE(6,865)
  106 READ(5,840)ADATA,IBLOCK,MOE,ABLOCK
      IF(IBLOCK)107,3160,107
 3160 WRITE(6,850)ABLOCK
      GO TO 106
  107 WRITE(6,845)ADATA,IBLOCK,MOE,ABLOCK
      IF(IBLOCK)180,106, 3170
 3170 MODS = 0
      IF(MOE-1) 109,3180,109
 3180 MODS=1
      WRITE(6,830)
  109 MOE = MODS
      NB = NR
      NEWBL(IBLOCK) = NEWBL(IBLOCK) + 1
      IF(MOE)115, 3200,115
 3200 ITEMS(IBLOCK) = 0
C---------GO TO SUBROUTINES TO READ IN BLOCKS ITEMS.
  115 GO TO(165,120,140,120,150,160,160,130,165,155,170,120),IBLOCK
  120 CALL THERM
      GO TO 105
C READ IN BLOCKS 2, 4, 12.
  130 CALL GEN
      GO TO 105
C READ IN BLOCK 8.
  140 CALL CHEM
      GO TO 105
C READ IN BLOCK 3.
  150 CALL FINK
      GO TO 105
C READ IN BLOCK 5.
  155 CALL FLOW
      GO TO 105
C READ IN BLOCK 10.
  160 CALL SURE
      GO TO 105
C READ IN BLOCKS 6 AND 7.
  165 CALL TALLY
      GO TO 105
C READ IN BLOCKS 1 AND 9.
  170 CALL PLOT
      GO TO 105
C READ IN BLOCK 11.
```

```
C---------INITIALIZE BEFORE FIRST TIME STEP.
  180 WRITE(6,865)
      WRITE(6,865)
      CALL CLOCK(CLOCKB,CLOCKA)
       CALL CLOCK1(KSEC,CLICKA)
      KSECS = KSEC
      KSECD=MOD(86400+KSEC-KSECS,86400)
      WRITE(6,825)CLOCKA,KSECD
  190 CALL TALLY
      IF(KWIT - 9) 3230,400, 3230
 3230 CALL SPECK
C INITIALIZE
C---------INCREMENT CYCLE COUNTER, DO NEXT TIME STEP.
  200 KCYC = KCYC + 1
C---------RETURN HERE AFTER A TIME STEP IS REJECTED.
CCC   GO TO SUBROUTINES TO DO TIME STEP CALCULATIONS AND WRITE OUT DATA.
  210 CALL THERM
      IF(NOGEN)230,230, 3240
 3240 CALL GEN
C HEAT SOURCES.
  230 IF(NKEM)240,240, 3250
 3250 CALL CHEM
C CHEM REACTION.
  240 IF(NOCON)245,245, 3260
 3260 CALL FINK
C INTERNAL FLUX.
  245 IF(NOFLOW)250,250, 3270
 3270 CALL FLOW
C MASS FLOW.
  250 IF(NOSCON)255,255, 3280
 3280 CALL SURE
C SURFACE NODES.
  255 IF(NMELT)260,260, 3290
 3290 CALL THERM1
C PHASE CHANGE.1
  260 IF(NOSPEC)270,270, 3300
 3300 CALL SPECK
C SPECIAL NODES.
  270 IF(NMELT)280,280, 3310
 3310 CALL THERM2
C PHASE CHANGE.2
C---------MAKE CRT PLOTS WHEN REQUIRED.
  280 IF(NUP)300,300, 3320
 3320 IF(NSTOP)500, 3330,290
 3330 IF((KCYC-2)*MOD(KCYC-1,JPIC))3340,3340,300
 3340  IF(KCYC)300,300, 3350
 3350 CONTINUE
  290 CALL PLOT
```

```
C MAKE PLOTS.                                                                   2390
  200 IF(NSTOP)500, 3360,400                                                    2400
 3360 IF(NDISC) 3370,310, 3370                                                  2410
 3370 KWIT = 6                                                                  2420
  310 CALL TALLY                                                                2430
       IF(KWIT)210,200, 3380                                                    2440
 3380 IF((KWIT - 5)*(KWIT - 9)) 3390,400, 3390                                  2450
 3390 NSTOP = 1                                                                 2460
      GO TO 200                                                                 2470
  400 KCYC = KCYC - 1                                                           2480
      WRITE(6,805)NPROB,KCYC.KWIT,SUMTIM                                        2490
      WRITE(6,806)                                                              2500
      IF(M - MW) 3490,480, 3490                                                 2510
 3490 WRITE(6,805)NPROB,KCYC,KWIT,SUMTIM                                        2520
  480 IF(NDISC) 3500, 3500,20                                                   2530
 3500 IF(NDUMP)500,500,500                                                      2540
C--------COME HERE WHEN JOB HAS BEEN COMPLETED.                                 2550
  500 WRITE(6,945)                                                              2560
C     SINCE COMP. WORK IS MADE WITH DISK (NOT TAPE) THE FOLLOWING
C     TWO STATEMENTS ARE CHANGED TO COMMENT. SEPT 1971
C     END FILE 9
C     REWIND 9
      IF(M - MW) 3510,520, 3510                                                 2590
 3510 WRITE(6,945)                                                              2600
  520 WRITE(6,945)                                                              2610
      STOP
      END                                                                       2620


      SUBROUTINE THERM                                                          2630
C     LIST 8                                                                    2640
C     CARDS COLUMN        THERM VERSION 5/29/68.                                2650
C     FORTRAN             THERM   TRUMP MATERIAL AND NODE SUB. FORTRAN-400      2660
CCC   VERSION  5/29/68.                                                         2670
      REAL*8 CON,DA,ZIP,TRAN                                                    2680
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                      2690
      REAL*8 FOR,SLIM,T,GEOM                                                    2700
      REAL*8 NAME,CAPS                                                          2710
      REAL*8 HMX,CTX,DAX,EX                                                     2720
      REAL*8 HMELT,A                                                            2730
      REAL*8 FORD                                                               2740
      REAL*8 NAMEL(3)
      DIMENSION APROP(8)                                                        2750
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
```

```
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPPOB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),F1(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      COMMON /ACHEMS/ KA(15),KAX(15),KB(15),KBX(15),
     1       B(355),BB(355),DB(355),DDB(355)
      DATA NAMEL/8H3        ,8H4        ,8HA=BLS    /
  800 FORMAT(A5,5I5,4E10.3,10A1)                                        3030
  806 FORMAT(/10X95HNAME MATL INDEX  KA  KB LTABC LTABK   DENSITY      C   3040
     *APACITY   CONDUCTIVITY   TMELT          HMELT)                     3050
  807 FORMAT(10X,A5,2I5,2I4,2I6,1P5E13.4)                               3060
  810 FORMAT (8E10.3)                                                   3070
  815 FORMAT(5I5,5X,5E10.3)                                             3080
  816 FORMAT(//,12X,22HNODE INDEX MATL NTYPE,4X,5HDLONG,10X,5HDWIDE,     3090
     $   10X,4HDRAD,11X,6HVOLUME)                                       3100
  817 FORMAT(10X,4I6,1P4E15.5)                                          3110
  825 FORMAT(10X,100(1H=))                                              3120
  828 FORMAT(10X,9HNODE DATA)                                           3130
  830 FORMAT(/,10X, 95HNODE  MATL NTYPE   RADIUS       VOLUME      MASS  3140
     $     CAPACITY  CONDUCTIVITY    ZIP          SLIM)                  3150
  835 FORMAT(10X,3I5,1X,7E12.4)                                         3160
  840 FORMAT(//,11X,29HNODP1 NODP2 INDEX NPROP  PROP)                   3170
  845 FORMAT(7I5)                                                       3180
  850 FORMAT(10X,4I6,2X,A6)                                             3190
  860 FORMAT(//,10X,16HMATERIAL SUMMARY,//,12X,92HNAME  MATL NODES   DEN 3200
     $SITY       CAPACITY      TOT VOL       TOT MASS      TOT CAP      TOT 3210
     $HEAT)                                                             3220
  870 FORMAT(10X,A6,2I6,1P6E13.5)                                       3230
```

```
  876 FORMAT(/17X12HCONDUCTIVITY5X5HSLOPE9X5HTVARK/,(15X1P3E15.6))
  880 FORMAT(10X,13HMATERIAL DATA,//,12X,10HNAME  MATL,5X,7HTOT CAP,
    1    8X,8HTOT HEAT,7X,8HAVG TEMP,7X,5HTMELT,10X,5HHMELT)
  885 FORMAT(/,10X,12HSYSTEM TOTAL,I6,26X,1P4E13.5)
  890 FORMAT(10X,A6,I6,1P5E15.5)
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)
  941 FORMAT(/19X65HCAPACITY        SLOPE                ENERGY            SLOPE
     $          TVARC/,(15X1P5E15.6))
      IF(KCYC) 3000,100,200
 3000 IF(IBLOCK - 2) 3010,10, 3010
 3010 IF(IBLOCK - 4) 3020,70, 3020
 3020 IF(IBLOCK - 12)7, 3030,7
 3030 CONTINUE
CARD BLOCK 12.  TABLE LOOK-UP CONNECTIONS.
      N= NTABS
      L = 0
      IF(MOE)1, 3040,1
 3040 NTABS=0
      N=NTABS
    1 READ(5,845)N1,N2,NP,NSEQ,NADP1,NADP2,NADPR
      IF(N1) 3050,7, 3050
 3050 CONTINUE
    2 IF(MOE) 3060,3,3
 3060 CALL SEEK2(N,N1,N2,NOXP1,NOXP2,NTABS,K)
      ITEMS(12)=MINO(N-1,ITEMS(12))
      GO TO 4
    3 NTABS=N+1
      N=NTABS
    4 IF(N - M12)5,5, 3070
 3070 NTABS=M12
      N=M12
      KWIT= 11
      WRITE(6,899)N,IBLOCK
    5 NOXP1(N)=N1
      NODP1(N)=NOXP1(N)
      NOXP2(N)=N2
      NODP2(N)=NOXP2(N)
      NPROP(N) =NP
      L= L + 1
      IF(MOD(L-1,57))6,3080,6
 3080 WRITE(6,840)
    6 WRITE(6,850)NODP1(N),NODP2(N),N,NPROP(N),APROP(NP)
      IF(NSEQ)1,1, 3090
 3090 NSEQ = NSEQ - 1
      N1 = N1 + NADP1
      N2 = N2 + NADP2
      NP = NP + NADPR
      GO TO 2
```

```
    7 RETURN
C COMPLETED BLOCK 12.
CARD BLOCK 2.  MATERIAL NAMES,NUMBERS,CHEMICAL COMPONENTS,THERM PROP.
   10 N= NMAT
      LABEL = 1
      IF(MOE)12, 3100,12
 3100 NMAT=0
      NVARK=NMAT
      NVARC=NVARK
      NMELT=NVARC
      N=NMELT
   12 READ(5,800)A1,N1,K1,K2,L1,L2,P1,P2,P3,P4,NX1
      IF(N1) 3110,68, 3110
 3110 IF(MOE) 3120,20,20
 3120 CALL SEEK1(N,N1,MAT,NMAT,K)
      IF(K)18,18, 3130
 3130 IF(HMELT(N)) 3140,14, 3140
 3140 NMELT = NMELT - 1
   14 IF(LTABC(N)) 3150,16, 3150
 3150 NVARC = NVARC - 1
   16 IF(LTABK(N)) 3160,18, 3160
 3160 NVARK = NVARK - 1
   18 ITEMS(2)=MINO(N-1,ITEMS(2))
      GO TO 22
   20 NMAT=N+1
      N=NMAT
   22 IF(N - M2)25,25, 3170
 3170 NMAT=M2
      N=NMAT
      KWIT = 11
      WRITE(6,899)N,IBLOCK
   25 AMAT(N) = A1
      MAT(N) = N1
      KAX(N)=K1
      KA(N)=KAX(N)
      KBX(N)=K2
      KB(N)=K2
      LTABC(N)=L1
      LTABK(N)=L2
      DENS(N) = P1
      CAPT(1,N) = P2
      CONT(1,N) = P3
      TMELT(N) = P4
      CALL PATCH(NX1,0.0,P5,LBX)
      HMELT(N)=ABS(P5)
      IF(LABEL)30,30, 3180
 3180 LABEL = 0
      WRITE(6,806)
```

```
   30 WRITE(6,807)AMAT(N),MAT(N),N,KA(N),KB(N),LTABC(N),
     $ LTABK(N),DENS(N),CAPT(1,N),CONT(1,N),TMELT(N),HMELT(N)
      IF(HMELT(N)) 3190,35, 3190
 3190 NMELT = NMELT + 1
      KB(N)=0
      KA(N)=KB(N)
   35 LTAB=IABS(LTABC(N))
      IF(LTAB - 2) 3200,38,38
 3200 LTABC(N) = 0
      GO TO 62
   38 NVARC = NVARC + 1
      LABEL = 1
      READ(5,810) (CAPT(J,N),TVARC(J,N),J=1,LTAB)
      DO 40 J = 2,LTAB
      SLOC(J,N) = (CAPT(J,N)-CAPT(J-1,N))/(TVARC(J,N)-TVARC(J-1,N))
   40 CONTINUE
      SLOC(1,N) = SLOC(2,N)
CCC   FIND ENERGY PEFERRED TO TVARC(1,N)
   42 IF(LTABC(N))60,60, 3210
 3210 WT(1,N) = 0.0
      DO 44 J = 2,LTAB
      F3(J) = 0.5*(CAPT(J,N) + CAPT(J-1,N))
      WT(J,N) = WT(J-1,N) + F3(J)*(TVARC(J,N) - TVARC(J-1,N))
   44 CONTINUE
      F3(1) = F3(2)
CCC   FIND CALCULATED ENERGY AT ZERO DEGREES
      DO 50 J= 1,LTAB
      J=J
      IF(TVARC(J,N)) 3220,54,52
 3220 CONTINUE
   50 CONTINUE
   52 CAPX = CAPT(J,N) - 0.5*SLOC(J,N)*TVARC(J,N)
      CORR = WT(J,N) - CAPX*TVARC(J,N)
      GO TO 56
   54 CORR = WT(J,N)
CCC   CALCULATE HEAT CONTENT RELATIVE TO ZERO DEGREES.
   56 DO 58 J= 1,LTAB
      WT(J,N) = WT(J,N) - CORR
   58 CONTINUE
   60 WRITE(6,941)(CAPT(J,N),SLOC(J,N),WT(J,N),F3(J),
     *    TVARC(J,N), J = 1,LTAB)
      IF(LTAB - M9)62,62, 3230
 3230 KWIT = 12
   62 LTAB=IABS(LTABK(N))
      IF(LTAB - 2) 3240,64,64
 3240 LTABK(N) = 0
      GO TO 12
   64 NVARK = NVARK + 1
```

```
       LABEL = 1
      READ(5,810) (CONT(J,N),TVARK(J,N),J=1,LTAB)
      DO 65 J = 2,LTAB
      SLOK(J,N) = (CONT(J,N)-CONT(J-1,N))/(TVARK(J,N)-TVARK(J-1,N))
   65 CONTINUE
      SLOK(1,N) = SLOK(2,N)
      WRITE(6,876)(CONT(J,N),SLOK(J,N),TVARK(J,N),J=1,LTAB)
      IF(LTAB-M9)12,12, 3250
 3250 KWIT = 12
      GO TC 12
   68 NEWBL(4) = NEWBL(4) + 1000
      RETURN
C COMPLETED BLOCK 2.
CARD BLOCK 4. NODE NUMBERS,MATERIALS,DIMENSIONS,INIT. T, A, B, G.
   70 N = NODES
      L= 0
      IF(MOE)74, 3260,74
 3260 NOSPEC=0
      NODES=NOSPEC
      N=NODES
   74 READ(5,815)N1,NSEQ,NADD,N2,KS,DLONG,DWIDE,DRAD
      IF(N1) 3270,94, 3270
 3270 ADD = 0.0
      IF(DRAD) 3280,75,75
 3280 ADD = -(DRAD + DRADS)
      DRAD = DRADS
   75 IF(MOE) 3290,80,80
 3290 CALL SEEK1(N,N1,NODE,NODES,K)
      IF(K)78,78,3300
 3300 IF(NTYPE(N)) 3310,78, 3310
 3310 NOSPEC =NOSPEC - 1
   78 ITEMS(4)=MINO(N-1,ITEMS(4))
      GO TO 82
   80 NODES=N+1
      N=NODES
   82 IF(N - M4)84,84, 3320
 3320 NODES=M4
      N=M4
      KWIT=11
      WRITE(6,899)N,IBLOCK
   84 NODE(N) = N1
      NOXMAT(N)=N2
      NODMAT(N)=NOXMAT(N)
      DO 86 J = 1,8
      NLOOK(N,J) = N
   86 CONTINUE
      T(N) = TONE
       A(N) = ALONE
```

```
        B(N) = BONE
        G(N) = GONE
       NTYPE(N) = 0
       DRAD=DRAD+ADD
       DRADS=DRAD
       RADIUS(N) = DRAD*SCALE
       VOL(N) = GEOM*DLONG*DWIDE*DRAD**KSYM*SCALE**3
       IF(VOL(N))88, 3330,88
  3330 VOL(N) = 1.0E-24
        NTYPE(N) = 2
        NOSPEC = NOSPEC + 1
        GO TO 89
    88 IF(KS) 3340, 89, 3340
  3340  NTYPE(N) = 3
        NOSPEC = NOSPEC + 1
    89 L = L + 1
       IF(MOD(L-1,57))90,3350,90
  3350 WRITE(6,816)
    90 WRITE(6,817)NODE(N),N,NODMAT(N),NTYPE(N),
      $ DLONG,DWIDE,DRAD,VOL(N)
       IF(NSEQ)74,74, 3360
  3360 NSEQ = NSEQ - 1
       N1 = N1 + NADD
       GO TO 75
    94 NX=NEWBL(7)
       DO 95 N = 5,12
       NEWBL(N) = NEWBL(N) + 1000
    95 CONTINUE
       NEWBL(7) = NX
       NEWBL(1) = NEWBL(1) + 1000
       RETURN
C COMPLETED BLOCK 4. RETURN TO HEART.
   100 IBLOCK = 4
       CALL REFER(NODMAT,NOXMAT,NODES,MAT,NMAT)
       IF(NTABS)140,140,3370
  3370 IBLOCK = 12
       CALL REFER(NODP1,NOXP1,NTABS,NODE,NODES)
       CALL REFER(NODP2,NOXP2,NTABS,NODE,NODES)
       IF(KWIT)600, 3380,600
  3380 DO 135 N = 1,NTABS
       J= NODP1(N)
       K= NPROP(N)
       NLOOK(J,K) = NODP2(N)
   135 CONTINUE
CCC    CALC. NODE MASSES, CAPACITIES, CONDUCTIVITIES.
   140 IF(KWIT)600, 3390,600
  3390 DO 150 N = 1,NODES
        J = NODMAT(N)
```

```
      ZIP(N) = 0.0
      HEFT(N) = VOL(N)*DENS(J)
      IF(HEFT(N)) 3400, 3400,142
 3400 HEFT(N) = 1.0E-36
  142 CAP(N) = HEFT(N)*CAPT(1,J)
      IF(CAP(N)) 3410, 3410,144
 3410 CAP(N) = 1.0E-36
  144 W(N) = CAP(N)*T(N)
      CON(N) = CONT(1,J)
      IF(CON(N)) 3420, 3420,150
 3420 CON(N) = 1.0E-24
  150 CONTINUE
CCC   FIND AND WRITE OUT TOTAL NODES, VOLUME, MASS, CAP OF EACH MATL.
      WRITE(6,860)
      HEFTS=0.0
      VOLS=0.0
      HEAT=0.0
      CAPS=0.0
      DO 170 K = 1,NMAT
      WMS=0.0
      CAPMS=WMS
      HEFTMS=CAPMS
      VOLMS=HEFTMS
      NODMS=VOLMS
      DO 160 N = 1,NODES
       IF(NODMAT(N) - K)160, 3430,160
 3430 NODMS = NODMS + 1
      VOLMS = VOLMS + VOL(N)
      CAPMS = CAPMS + CAP(N)
      HEFTMS = HEFTMS + HEFT(N)
      WMS = WMS + W(N)
  160 CONTINUE
      CAPS = CAPS + CAPMS
      VOLS = VOLS + VOLMS
      HEFTS = HEFTS + HEFTMS
      HEAT = HEAT + WMS
      IF(CAPMS)170,170, 3440
 3440 WRITE(6,870)AMAT(K),MAT(K),NODMS,DENS(K),
     $CAPT(1,K),VOLMS,HEFTMS,CAPMS,WMS
  170 CONTINUE
      WRITE(6,885)NODES,VOLS,HEFTS,CAPS,HEAT
CCC  START EACH NEW PROBLEM  WITH PHASES IN EQUILIBRIUM.
      IF(NMELT) 3450,200,3450
 3450 IF(1.-0.)200,3451,200
 3451 IF(NAME(16)-NAMEL(1))3455,200,3455
 3455 IF(NAME(16)-NAMEL(2))3460,200,3460
 3460 IF(NAME(15)-NAMEL(3))3470,200,3470
 3470 CONTINUE
```

```
C COLUMNS 67-71, RETAIN BLOCK 9 A.
      DO 190 N = 1,NODES
       J = NODMAT(N)
      IF(HMELT(J)) 3480,190, 3480
 3480 A(N) = 1.0
      IF(T(N)-TMELT(J))190,190, 3490
 3490 A(N) = 0.0
  190 CONTINUE
  200 FORD = FOR*DELT
      IF(KWIT)215, 3500, 3500
 3500 IF(NOW)215,215, 3510
 3510 IF(KCYC - 1)215,202, 3520
 3520  IF(KDATA)215, 3530, 3530
 3530 CONTINUE
CCC   FIND, WRITE TOTAL CAP, HEAT CONTENT, AVG TEMP OF EACH MATL.
  202 WRITE(6,880)
      DO 210 K = 1,NMAT
      TMS=0.0
      WMS=TMS
      CAPMS=WMS
      DO 205 N = 1,NODES
       IF(NODMAT(N)-K)205, 3540,205
 3540 CAPMS = CAPMS + CAP(N)
      WMS= WMS + W(N)
      TMS = TMS + CAP(N)*T(N)
  205 CONTINUE
      IF(CAPMS)210,210, 3550
 3550 TMS = TMS/CAPMS
      WRITE(6,890)AMAT(K),MAT(K),CAPMS,WMS,TMS,
     1   TMELT(K),HMELT(K)
  210 CONTINUE
      WRITE(6,825)
CCC   INITIALIZE HEAT CONTENT, CHANGES IN T, A, B AND FLUX EACH CYCLE.
  215 DO 220 N = 1,NODES
      DF(N)=0.0
      DT(N)=0.0
      DB(N)=0.0
      DA(N)=0.0
      KOP1=NODMAT(N)
      IF(LTABC(KOP1))220,3560,220
 3560 W(N)=CAP(N)*T(N)+HMELT(KOP1)*HEFT(N)*(1.0-A(N))
  220 CONTINUE
CCC   WRITE OUT DATA ON PRINT-OUT CYCLES.
      IF(NOW)228,228, 3570
 3570  IF(KCYC - 1)228,222, 3580
 3580  IF(KDATA)228,228, 3590
 3590 IF(NVARC + NVARK + NVARH + NORAD + NORADS)228,228, 3600
 3600 CONTINUE
```

```
CCC    WRITE OUT PROPERTIES OF EACH NODE.
  222  WRITE(6,828)
       DO 225 N = 1,NODES
       IF(MOD(N-1,57))224,3610,224
 3610  WRITE(6,830)
  224  WRITE(6,835)NODE(N),NOXMAT(N),NTYPE(N),RADIUS(N),
     1    VOL(N), HEFT(N), CAP(N), CON(N), ZIP(N), SLIM(N)
  225  CONTINUE
       WRITE(6,825)
CCC    FIND NEW NODE HEAT CAPACITIES, HEAT CONTENT
  228  IF(NVARC)300,300, 3620
 3620  DO 280 N = 1,NODES
        J = NODMAT(N)
        K = NLOOK(N,1)
       IF(LTABC(J)) 3630,280,230
 3630  SET = SUMTIM + FORD
       GO TO 235
  230  EX = FORD*DDT(K)
       SET = T(K) + EX
  235  CAPS = CAPS - CAP(N)
        BET = CAP(N)
       BETW = W(N)
       MIN = 1
       MAX=IABS(LTABC(J))
  240  MID = (MIN + MAX)/2
        IF(SET - TVARC(MID,J))250,270,260
  250  MAX = MID
        IF(MAX - 2)270,240,240
  260  MIN = MID
        IF(MAX - MIN - 2)270,240,240
  270  CAPX = CAPT(MID,J) + SLOC(MID+1,J)*(SET - TVARC(MID,J))
       CAP(N) = HEFT(N)*CAPX
       IF(CAP(N)) 3640, 3640,272
 3640  CAP(N) = 1.0E-36
  272  IF(LTABC(J))275,280, 3650
 3650  CAPX = 0.5*(CAPT(MID,J) + CAPX - SLOC(MID+1,J)*EX)
       W(N) = HEFT(N)*(WT(MID,J) + CAPX*(SET - EX - TVARC(MID,J)))
       W(N) = W(N) + HEFT(N)*HMELT(J)*(1.0 - A(N))
  275  CAPS = CAPS + CAP(N)
       BETW=ABS(BETW-W(N))/BET
       BET=100.0*DABS(BET-CAP(N))/BET
       DTMAX=AMAX1(DTMAX,BET*TVARY)
       DTMAX=AMAX1(DTMAX,BETW)
  280  CONTINUE
CCC    FIND NEW NODE THERMAL CONDUCTIVITIES.
  300  IF(NCCON*NVARK)600,600, 3660
 3660  DO 380 N = 1,NODES
        J = NODMAT(N)
```

```
      K = NLOOK(N,2)
      IF(LTABK(J)) 3670,380,320
 3670  SET = SUMTIM + FORD
      GO TO 330
  320 SET = T(K) + FORD*DDT(K)
  330 MIN = 1
      MAX=IABS(LTABK(J))
      BET = CON(N)
  340 MID = (MIN + MAX)/2
       IF(SET - TVARK(MID,J))350,370,360
  350 MAX = MID
       IF(MAX - 2)370,340,340
  360 MIN = MID
       IF(MAX - MIN - 2)370,340,340
  370 CON(N) = CONT(MID,J) + SLOK(MID+1,J)*(SET - TVAPK(MID,J))
      IF(CON(N)) 3680, 3680,375
 3680 CON(N) = 1.0E-24
  375 SCON=CON(N)
      BET=100.0*ABS(BET-SCON)/BET
      DTMAX=AMAX1(DTMAX,BET*TVARY)
  380 CONTINUE
  600 RETURN
      ENTRY THERM1
      IF(KWIT) 3690,3690,600
 3690 CONTINUE
CCC   ESTIMATE EFFECT OF PHASE CHANGE IN NODE TEMPS,SAVE AS DDA.
      DO 660 N= 1,NODES
       J = NODMAT(N)
       IF(HMELT(J)) 3700,660, 3700
 3700 IF(NTYPE(N) - 2)610, 3710,610
 3710 DDA(N) = 0.0
      GO TO 660
  610 HMX = HEFT(N)*HMELT(J)/CAP(N)
      DTX = T(N) + DELT*(DDT(N) - HMX*DDA(N)) - TMELT(J)
      DDA(N)=HMX*DMAX1(-A(N),DMIN1(-DTX/HMX,1.0-A(N)))
      DT(N) = DT(N) + DDA(N)
  660 CONTINUE
       RETURN
C COMPLETED THERM1.  RETURN TO HEART.
      ENTRY THERM2
      IF(KWIT) 3720,3720,600
 3720 CONTINUE
CCC   FIND EFFECT OF PHASE CHANGE ON CONCENTRATION AND TEMP IN NODES.
      DO 760 N = 1,NODES
       J = NODMAT(N)
       IF(HMELT(J)) 3730,760, 3730
 3730 DT(N) = DT(N) - DDA(N)
      DDA(N) = 0.0
```

```
      DTX = T(N) + DT(N) - TMELT(J)                                         07530
      HMX = HEFT(N)*HMELT(J)/CAP(N)                                         07540
  710 IF(NTYPE(N) - 2)740, 3740,740                                         07550
 3740 HMX=1.0E-06*DABS(DTX)                                                 07560
  740 DAX=DMAX1(-A(N),DMIN1(-DTX/HMX,1.0-A(N)))                             07570
      IF(DAX) 3750,760, 3750                                                07580
 3750 ABDT=DABS(DT(N))                                                      07590
      DTMAX=AMAX1(DTMAX,ABDT)                                               07600
      DT(N) = DT(N) + DAX*HMX                                               07610
      DA(N) = DA(N) + DAX                                                   07620
  760 CONTINUE                                                              07630
      RETURN                                                                07640
C COMPLETED THERM2. RETURN TO HEART.                                        07650
      END                                                                   07660


      SUBROUTINE TALLY                                                      07670
C     CARDS COLUMN      TALLY STD VERSION 5/29/68                           07680
C     LIST 8                                                                07690
C     FORTRAN           TALLY   TRUMP CALC. CONTROL SUB.      FORTRAN-400   07700
CCC   STD VERSION 5/29/68.  (DO NOT USE WITH TRUMP/DS VERSION.)             07710
CCC   5/14/68  ADDED TEST AT 225 TO COMPARE SMALL WITH SMALT.               07720
CCC   5/29/68 CHANGED NSTORE CALC AND FORMAT TO CORRECT FOR REMOVAL OF      07730
CCC           ARRAY SLOW(M9,12) FROM COMMON ATHERM.                         07740
      REAL*8 HMELT,A                                                        07750
      REAL*8 HMX                                                            07760
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                  07770
      REAL*8 CON,DA,ZIP,TRAN                                                07780
      REAL*8 NAME,CAPS                                                      07790
      REAL*8 FOR,SLIM,T,GEOM                                                07800
      REAL*8 TMM,CNN                                                        07810
      REAL*8 DFFF(355)                                                      07820
      COMMON /CURE/ EQCURE(355)
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKFM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
```

```
  3         TMAX,TMIN,TONE,TVARY
    COMMON  HMELTX(15),LTABK(15),
  1         NLOOK(355,8),NODE(355),
  2         NODMAT(355),NTYPE(355),RADIUS(355),
  3         VOL(355),W(355),
  4         DFI(950),FI(950),NOD1(950),NOD2(950),
  5         F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
    COMMON  AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
  1         LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
  2         TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
  3  NOXMAT(355),
  4         NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
    COMMON  AA(355),F(355),GG(355),H(355),NOTE(355),
  1         NOXE(355),NTYPES(355),TT(355)
    COMMON  /ACHEMS/ KA(15),KAX(15),KB(15),KBX(15),
  1         B(355),BB(355),DD(355),DDF(355)
803 FORMAT(/10X5HCYCLEI5,10H MADE NODEI5,15H \ SPECIAL NODE)                08110
805 FORMAT(//10X,25HINPUT ERROR, MATERIALS =,I5,10H, NODES = ,I5,1H.)       08120
810 FORMAT (8E10.3)                                                         08130
811 FORMAT(//12X89H TONE          ALONE          BONE          GONE       FO 08140
   $NE        HONE          RONE          PONE/10X1P8E12.4)                  08150
812 FORMAT(1H1,9X,17HTRUMP OUTPUT DATA,69X,9HDATA DECK,I5,//,              08160
  1    10X,A1,14A5,A1,4A2,4X,2(2X,A6))                                      08170
815 FORMAT(/,10X,59H PRINTOUT    CYCLE   TOO FAST  TOO SLOW      KWIT       08180
   $    DELTMX,10X,5HSMALL,11X,5HTVARY,6X,4HNUTS,/,7X,5I10,1P3E16.5,I5)     08190
820 FORMAT(/,10X,87H   TOTAL TIME     TIME STEP     HEAT FLOW     TEMP      08200
   $ FROM FLUX   FLUX RATE      TEMP RATE,/,8X,1P6E15.5,//,                 08210
   *10X,89H   AVG TEMP     HEAT CAPACITY   HEAT CONTENT    GEN RATE        08220
   $    HEAT GEN      TEMP FROM GEN,/,8X,1P6E15.5)                          08230
830 FORMAT(28HBLOCK 9.  T, A, B, G AT TIME,20X,1PE13.5,5X,I6)               08240
835 FORMAT (3I5,5X,40A1)                                                    08250
836 FORMAT(//,12X,10HNOTE INDEX,10X,2HTT,13X,2HAA,13X,2HPB,13X,2HGG)        08260
837 FORMAT(10X,2I6,4X,1P4E15.6)                                             08270
840 FORMAT(I5,15X,1P4E10.3)                                                 08280
850 FORMAT(2H**,5X,A1,5A5,15X,2A6)                                          08290
855 FORMAT(9I5,15X,E10.3)                                                   08300
860 FORMAT (2I5,7E10.3)                                                     08310
861 FORMAT(//11X,95H KD    KT      DELTO          SMALL          TVARY      08320
   $  TAU          TIMAX          TMIN          TMAX,/,10X2I5,1P7E13.5)     08330
862 FORMAT(//10X45H     KD KSYM    GEOM          SIGMA          TBASE,/,    08340
   $ 10X2I5,1P3E13.5)                                                       08350
865 FORMAT(10X,12HFINAL HEAT =,1PE15.5,5X,12HFINAL FLOW =,E15.5)            08360
870 FORMAT(//10X,54HIPRINT   NUM KDATA KSPEC  MCYC  MSEC NPUNCH NDOT I      08370
   1RITE,22X,5HSCALE,/,10X,9I6,18X,E13.5)                                   08380
875 FORMAT(//,10X,94H  NODE      TEMP          NODE    TEMP         NODE    08390
   $ TEMP         NODE    TEMP          NODE     TEMP,/,(10X,I6,1PE14.5,    08400
   $ I6,E14.5,I6,E14.5,I6,E14.5,I6,E14.5))                                  08410
880 FORMAT(/12X,93HNODE          TEMP          W/F A         DA          D  08420
```

```
      $DA             W/F B          DB                 DDB)                                08430
  882 FORMAT(10X,I6,3X,1P7E13.4)                                                             08440
  885 FORMAT(/7X,112HNODE          TEMP          DT                DDT          GE           08450
     1N RATE          W              H             F          CURE AT 280 F)                 08460
  886 FORMAT(5X,I6,3X,8E13.4)                                                                08470
  890 FORMAT(5X,115(1H=))                                                                    08480
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)                          08490
  900 FORMAT(/,26X,31HMAXIMUM ALLOWED TABLE LENGTH IS,I4,1H.)                                08500
  905 FORMAT(/,10X,94HARRAY STORAGE = 3*M11+M1*(1+M11)+M2*(11+7*M9)+M3*(                     08510
     15+9*M9)+56*M4+12*M5+M6*(12+3*M9)+M7*(5+3*M9),/,                                        08520
     2   26X,38H+M8*(3+3*M9)+M10*(9+3*M9)+5*M12+3*M9 =,I8,1H.)                               08530
  910 FORMAT(/,10X,6HTABLES,10X,60H  CAPT  CONT     QT     ZT      ET HSURT                  08540
     1TEMPB     GT FLOWT TOTAL,/,26X,10I6)                                                   08550
  915 FORMAT(//10X,21HSUMMARY OF INPUT DATA,/10X12HBLOCK NUMBER,4X11I6)                      08560
  920 FORMAT(10X,12HMAXIMUM SIZE,4X,11I6)                                                    08570
  925 FORMAT(10X,12HINITIAL SIZE,4X,11I6)                                                    08580
  926 FORMAT(10X,15HUNMODIFIED SIZE,1X,11I6)                                                 08590
  930 FORMAT(10X,12HFINAL SIZE  ,4X,11I6)                                                    08600
  935 FORMAT(10X,12HITEM NAME   ,4X,66H  MAT   KEM  NODE  NOD1  NODS  N                      08610
     1ODB  NODG  NOTE NODF1 NODEP NODP1)                                                     08620
  940 FORMAT(/10X,12HOTHER TOTALS,4X,38HNOSPEC NOGEN NORAD NORADS NMELT                      08630
     1NRFACT,/,26X,6I6)                                                                      08640
  945 FORMAT(10X,13HTIMES READ IN,3X,11(2X,I4))                                              08650
  950 FORMAT(10X,17HWILL REPEAT CYCLE,I6,4X,8H DTMAX =,1PE10.3,                              08660
     1   8H DTPRE =,E10.3,7H DELT =,E10.3,9H SUMTIM =,E10.3)                                 08670
  955 FORMAT(5(1H*),4HNODE,I6,2X,16HT,DDT,A,B,G,EO =,1P6E12.4,                               08680
     1     1X,7HAT TIME,E12.4)                                                               08690
      IF(KCYC) 3000,160,200                                                                  08700
 3000 IF(IBLOCK)100,3010,30                                                                  08710
 3010 WRITE(6,915) (I,I=2,12)                                                                08720
      WRITE(6,935)                                                                           08730
      WRITE(6,925) (ITEMS(I),I=2,12)                                                         08740
      NIT = NODES                                                                            08750
      IF(NOW - 2)15,3020,15                                                                  08760
 3020 CONTINUE                                                                               08770
C NOW = NAME(16)                                                                             08780
CCC   RESTORE NODE TYPE TO INITIAL VALUE OF PREVIOUS PROBLEM.                                08790
      DO 10 N = 1,NODES                                                                      08800
      IF(NTYPE(N)-NTYPES(N)) 3030,10, 3030
 3030 NTYPE(N)=NTYPES(N)                                                                     08820
      NOSPEC=NOSPEC-1                                                                        08830
   10 CONTINUE                                                                               08840
      GO TO 25                                                                               08850
CCC   CHANGE INTERNAL BLOCK 9 TO FINAL T, A, B, OF PREVIOUS PROBLEM.                         08860
   15 DO 20 N = 1,NODES                                                                      08870
      TT(N) = T(N)                                                                           08880
      AA(N) = A(N)                                                                           08890
      BB(N) = B(N)                                                                           08900
```

```
   20 CONTINUE
   25 RETURN
   30 IF(IBLOCK - 9) 3040,91, 3040
 3040 IF(IBLOCK - 1)400, 3050,400
 3050 CONTINUE
CARD BLOCK 1. PROBLEM LIMITS, CONTROLS, CONSTANT I.C. AND B.C.
CCC   IRITE ONLY USED IN TRUMP/6600/DS VERSION.
      NIT=0
      READ(5,855) IPRINT,NUM,KDATA,KSPEC,MCYC,
     $  MSEC,NPUNCH,NDOT,IRITE,SCALE
      IPRINT=MAXO(1,IPRINT)
      NUMX = NUM
      KDATAX = KDATA
      IF(SCALE) 3060, 3060,32
 3060 SCALE = 1.0
   32 IF(MCYC) 3070, 3070,36
 3070 MCYC = 30000
   36 IF(MSEC) 3080, 3080,40
 3080 MSEC = 30000
   40 WRITE(6,870) IPRINT,NUM,KDATA,KSPEC,MCYC,MSEC,
     1   NPUNCH,NDOT,IRITE,SCALE
      READ(5,860) KD,KT,DELTO,SMALL,TVARY,TAU,TIMAX,TMIN,TMAX
      KD=MAXO(1,KD)
      KT=MAXO(1,KT)
      KSYM = (KD + 3)/3
      GEOM = 2.0**(KD - 1)*3.1415926**(KD/2)
      IF((DELTO -1.0E-10)*(DELTO - 1.0E12))50,50,3090
 3090  DELTO = 1.0E12
   50 SMALL=AMAX1(SMALL,1.0E-12)
       SMALT = SMALL
      IF(TMAX - TMIN) 3100, 3100, 60
 3100  TMAX = 1.0E12
       TMIN =-1.0E12
   60 IF(TIMAX) 3110, 3110, 65
 3110  TIMAX = 1.0E06*DELTO
   65 IF(TVARY) 3120, 3120, 70
 3120 TVARY=AMIN1(5.0,0.005*(TMAX-TMIN))
   70 WRITE(6,861)KD,KT,DELTO,SMALL,TVARY,TAU,TIMAX,TMIN,TMAX
      TMIN = TMIN - 0.001*TVARY
       TMAX = TMAX + 0.001*TVARY
      GO TO(81,82,83,84,85),KT
   81 TBASE = 273.15
       SIGMA = 1.355E-12
      GO TO 90
   82 TBASE = 0.0
       SIGMA = 1.355E-12
      GO TO 90
   83 TBASE = 460.0
```

```
       SIGMA = 0.173E-8                                                 09390
      GO TO 90                                                          09400
   84 TBASE = 0.0                                                       09410
       SIGMA = 0.173E-8                                                 09420
      GO TO 90                                                          09430
   85 TBASE = 0.0                                                       09440
       SIGMA = 1.0                                                      09450
   90 WRITE(6,862)KD,KSYM,GEOM,SIGMA,TBASE                              09460
      READ(5,810)TONE,ALONE,BONE,GONE,FONE,HONE,RONE,PONE               09470
      WRITE(6,811)TONE,ALONE,BONE,GONE,FONE,HCNE,RONE,PONE              09480
      RETURN                                                            09490
C COMPLETED BLOCK 1. RETURN TO HEART.                                   09500
CAPD BLOCK 9. INITIAL VALUES OF T, A, B, AND G.                         09510
   91 N = NIT                                                           09520
      L = 0                                                             09530
      IF(MOE)92, 3130,92                                                09540
 3130 NIT=0                                                             09550
      N=0                                                               09560
   92 READ(5,835)N1,NSEQ,NADD,NX1,NX2,NX3,NX4                           09570
      IF(N1) 3140,98, 3140                                              09580
 3140 CALL PATCH(NX1,TONE,TX,LXX)                                       09590
      CALL PATCH(NX2,ALONE,AX,LXX)                                      09600
      CALL PATCH(NX3,BONE,BX,LXX)                                       09610
      CALL PATCH(NX4,GONE,GX,LXX)                                       09620
   93 IF(MOE) 3150,94,94                                                09630
 3150 CALL SEEK1(N,N1,NOXE,NIT,K)                                       09640
      ITEMS(9)=MIN0(N-1,ITEMS(9))                                       09650
      GO TO 95                                                          09660
   94 NIT=N+1                                                           09670
      N=N+1                                                             09680
   95 IF(N - M4)96,96, 3160                                             09690
 3160 NIT=M4                                                            09700
      N=M4                                                              09710
      KWIT= 11                                                          09720
      WRITE(6,899)N,IBLOCK                                              09730
   96 NOXE(N)=N1                                                        09740
      NOTE(N)=N1                                                        09750
      TT(N)  =  TX                                                      09760
      AA(N)  =  AX                                                      09770
      BB(N)  =  BX                                                      09780
      GG(N)  =  GX                                                      09790
      L = L + 1                                                         09800
      IF(MOD(L-1,57))97,3170,97                                         09810
 3170 WRITE(6,836)                                                      09820
   97 WRITE(6,837)NOTE(N),N,TT(N),AA(N),BB(N),GG(N)                     09830
      IF(NSEQ)92,92, 3180                                               09840
 3180 NSEQ = NSEQ - 1                                                   09850
      N1 = N1 + NADD                                                    09860
```

```
      GO TO 93
   98 RETURN
C COMPLTED BLOCK 9.
  100 WRITE(6,915)(I,I=2,12)
      WRITE(6,945)(NEWBL(I),I=2,12)
      WRITE(6,935)
      WRITE(6,920)M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12
      WRITE(6,926)(ITEMS(I),I=2,12)
      NTABLE =NVARC+NVARK + NVARQ+NVARZ+NVARE + NVARH+NVART+NVARG+NVARFL
      WRITE(6,910)NVARC,NVARK,NVARQ,NVARZ,NVARE,NVARH,
     1   NVART,NVARG,NVARFL,NTABLE
      WRITE(6,900)M9
      IF(NMAT)102,102, 3190
 3190 IF(NODES) 3200,3200,105
 3200 CONTINUE
  102 KWIT = 9
C     IN FOLLOWING WRITE STATEMENT LOGICAL UNIT NUMBER 3 IS
C     CHANGED TO 6.SEPT 1971
      WRITE(6,805)NMAT,NODES
      RETURN
C MISSING BLOCK 2 OR 4.  RETURN TO HEART.
  105 IF(NUM) 3210,108, 3210
 3210 IBLOCK = 1
      CALL REFER(NUM,NUMX,1,NODE,NODES)
  108 IF(NIT)115,115, 3220
 3220 IBLOCK = 9
      CALL REFER(NOTE,NOXE,NIT,NODE,NODES)
      IF(KWIT)130, 3230,130
 3230 CONTINUE
C ASSIGN INITIAL CONDITIONS TO NODES.
      DO 110 N = 1,NIT
      J = NOTE(N)
      T(J) = TT(N)
      A(J) = AA(N)
      B(J) = BB(N)
      G(J) = GG(N)
  110 CONTINUE
CCC   SET UP INTERNAL BLOCK 9 WITH INITIAL CONDITIONS OF THIS PROBLEM.
  115 DO 120 N = 1,NODES
      NOTE(N) = N
      NOXE(N) = NODE(N)
      TT(N) = T(N)
      AA(N) = A(N)
      BB(N) = B(N)
      GG(N) = G(N)
      NTYPES(N) = NTYPE(N)
  120 CONTINUE
C MAKE ALL NODES SPECIAL IF KSPEC IS POSITIVE.
```

```
      IF(KSPEC)130,130, 3240
 3240 DO 125 N= 1,NODES
      IF(NTYPE(N))125, 3250,125
 3250 NTYPE(N) = 5
      NOSPEC = NOSPEC + 1
  125 CONTINUE
CCC  INITIALIZE  BEFORE  FIRST TIME INCREMENT.
  130 NOGEN = 0
      NREACT = 0
      DO 135 N = 1,NODES
      H(N)=0.0
      F(N)=0.0
      IF(A(N))132, 3260,132
 3260 IF(B(N)) 3270,134, 3270
 3270 CONTINUE
  132 NREACT = NREACT +1
  134 IF(G(N)) 3280,135, 3280
 3280 NOGEN = NOGEN + 1
  135 CONTINUE
      NSTORE = 3*M11+M1*(1+M11)+M2*(11+7*M9)+M3*(5+9*M9)+56*M4+12*M5+
     1   M6*(12+3*M9)+M7*(5+3*M9)+M8*(3+3*M9)+M10*(9+3*M9)+5*M12+3*M9
      WRITE(6,905)NSTORE
      WRITE(6,940)NOSPEC,NOGEN,NORAD,NORADS,NMELT,NREACT
      NREACT = NREACT + NMELT + NKEM
      NOGEN = NOGEN + NVARG
CCC   INITIALIZE BEFORE FIRST TIME STEP (KCYC = 0, DELT = 1.0E-12).
      SUMTIM = TAU
      DELT = 1.0E-12
      DTMAX=0.0
      DELTSS=0.0
      DELTS=0.0
      DELTMX=AMAX1(1.0E-10,DELTO)
      FOR = 1.0
      GO TO 710
  160 IF(KWIT) 3290, 3290,750
 3290 CONTINUE
C INITIALIZE AT END OF CYCLE 0.
      MF = 0
       MS = 0
       KGOOD = 0
       NPRINT = 0
      KDATA = KDATAX
      DTMAX = 0.
       GENS = 0.
       RAT1 = 1.0
       BIG= 0.99999999E12
      GO TO 205
CCC   START HERE WHEN KCYC = 1 OR MORE, TALLY RESULTS OF CALCULATIONS.
```

```
  200 IF(KCYC - 1) 3300, 3300,202                                                         14490
 3300 DTMAX = 0.                                                                          14500
  202 IF(KWIT) 3310, 3310,410                                                             14510
 3310  KWIT = 0                                                                           14520
       NOW = 0                                                                            14530
      IF(BIG - DELTMX)225,225, 3320                                                       14540
 3320 CONTINUE                                                                            14550
C FIND MAX STABLE TIME STEP.                                                               14560
      IF(KCYC - 1)205,205, 3330                                                           14570
 3330  IF(NTABLE) 3340, 3340,205
 3340  IF(KSPEC)225, 3350, 3350                                                           14590
 3350 CONTINUE                                                                            14600
  205 DELTMX = 1.5E12                                                                     14610
      NREG =0                                                                             14620
      DO 210 N = 1,NODES                                                                  14630
      SLIM(N) = 1.E24                                                                     14640
      IF(ZIP(N))208,208, 3360                                                             14650
 3360 CAPZIP=CAP(N)/ZIP(N)                                                                14660
      SLIM(N)=AMAX1(1.0E-24,CAPZIP)                                                       14670
  208 IF(NTYPE(N))210, 3370,210                                                           14680
 3370 NREG=NREG+1                                                                         14690
      SSLIM=SLIM(N)                                                                       14700
      DELTMX=AMIN1(DELTMX,SSLIM)                                                          14710
  210 CONTINUE                                                                            14720
      DELTMX=AMAX1(1.0E-10,DELTMX/1.5)                                                    14730
      IF(DELTMX - DELTO)215,225, 3380                                                     14740
 3380 DELTMX = DELTO                                                                      14750
      GO TO 225                                                                           14760
CCC   CHANGE NODES TO SPECIAL NODES IF NECESSARY TO INCREASE DELTMX.                      14770
  215 IF(MS)225,225, 3390                                                                 14780
 3390 IF(NREG)225,225, 3400                                                               14790
 3400 IF(KSPEC)225, 3410, 3410                                                            14800
 3410 NEWS = 0                                                                            14810
      DO 220 N = 1,NODES                                                                  14820
      IF(NTYPE(N))220, 3420,220                                                           14830
 3420 IF(1.8*DELTMX-SLIM(N))220, 3430, 3430                                               14840
 3430 NTYPE(N)=4                                                                          14850
      NOSPEC=NOSPEC+1                                                                     14860
      NEWS=1                                                                              14870
      WRITE(6,803)KCYC,NODE(N)                                                            14880
  220 CONTINUE                                                                            14890
      IF(NEWS)225,225,3440                                                                14900
 3440 KGOOD = 0                                                                           14910
      MS = 0                                                                              14920
      GO TO 205                                                                           14930
CCC   RESTORE SMALL TO SMALT,IF SMALL HAS DECREASED BELOW SMALT.                          14940
  225 IF(SMALT - SMALL)226,226, 3450                                                      14950
 3450 SMALL = SMALT                                                                       14960
```

```
CCC    CALC SMALL IF NOT READ IN, DELTMX NOT 1.E12, NREG NOT 1/4 NODES.
  226 IF(SMALL - DELTMX)228, 3460, 3460
 3460 SMALL = 0.99999999*DELTMX
  228 IF(1.0 - BIG*SMALT)230, 3470, 3470
 3470  IF(4*NREG - NODES)230, 3480, 3480
 3480 IF(BIG - DELTMX)230, 3490, 3490
 3490  SMALL = 0.01*DELTMX
CCC    FIND LARGEST TEMPERATURE CHANGE AMONG ALL BUT ZERO-VOLUME NODES.
  230 DTPRE = DTMAX
       DO 235 N = 1,NODES
      IF(KCYC - 1) 3500, 3500,232
 3500 IF(NTYPE(N) - 2) 3510,235, 3510
 3510 CONTINUE
  232 DABDT=DABS(DT(N))
      DTMAX=AMAX1(DTMAX,DABDT)
  235 CONTINUE
      RATIO=TVARY/AMAX1(1.0E-24,DTMAX)
CCC   REPEAT CYCLE IF MAX TEMP CHANGE MORE THAN DOUBLE TVARY.
      IF(RATIO - 0.5) 3520,240,240
 3520  IF(DELT - 1.2*SMALL)240,240, 3530
 3530 WRITE(6,950)KCYC,DTMAX,DTPRE,DELT,SUMTIM
      KWIT = -1
       KGOOD = 0
       GO TO 245
  240 SUMTIM = SUMTIM + DELT
       KGOOD = KGOOD + 1
  245 DELTSS = DELTS
      DELTS = DELT
      DTMAX = 0.
      RAT2 = RAT1
       RAT1= RATIO
CCC   START OFF WITH SMALL TIME INCREMENT WITH NO REG NODES.
      IF(KCYC) 3540, 3540,248
 3540 IF(NREG) 3550, 3550,260
 3550 RATIO = 0.01*RATIO
      GO TO 260
CCC   CHANGE DELT TO MAKE NEXT DTMAX CLOSER TO TVARY.
  248 IF(RATIO-1.0) 3560,260,250
 3560 RATIO=AMAX1(0.5,RATIO*RATIO)
      GO TO 260
  250 RATIO = AMIN1( 2.0,0.5*(1.0 + RATIO))
  260 DELT = DELTS*RATIO
       DELTSX = DELTS
      IF(KGOOD - 2)270, 3570, 3570
 3570  IF(RAT1 - RAT2)270, 3580,3580
 3580 DELTSX = DELT/((RAT1/RAT2)**(-DELTS/DELTSS) + 1.0E-24)
CCC   FIND INTERPOLATION FACTOR FOR NEXT TIME STEP.
  270 FOR=AMAX1(0.57,AMAX1(1.0,PAT1)/(1.0+RAT1))
```

```
      IF(KSPEC - 2)271, 3590, 3590
 3590 FOR = 1.0
      IF(KSPEC - 3)272, 3600, 3600
 3600 FOR = 0.5
      GO TO 272
  271 IF(DELT - BIG)272, 3610, 3610
 3610 FOR = 1.0
CCC   KEEP TIME STEP IN RANGE FROM SMALL TO DELTMX OR DELTO.
  272 IF(DELT - SMALL) 3620,275,275
 3620 DELT = SMALL
      MF = MF + 1
      GO TO 280
  275 IF(DELT - DELTMX)280,280, 3630
 3630  DELT = DELTMX
       MS = MS + 1
  280 IF(KWIT) 3640,300,300
 3640  RETURN
C REPEAT CYCLE WITH NEW DELT.
CCC   FIND NEW TEMPERATURES, HEAT CONTENTS,AND FLUXES.
  300 GS = 0.0
      DO 301 K=1,NODES
  301 DFFF(K)=DF(K)
       TMAX1 = -1.E8
       TMIN1 = 1.E8
      CUF=DELTS*60.0
      DO 310 N = 1,NODES
      T(N) = T(N) + DT(N)
C     EQCURE CALCULATION IS CHANGED TO COMMENT .SEPT 1971
C     EQCURE(N)=EQCURE(N)+2.0**((T(N)-280.0)/18.0)*CUF
      GS = GS + G(N)
      SIPT=T(N)
      TMAX1=AMAX1(TMAX1,SIPT)
      TMIN1=AMIN1(TMIN1,SIPT)
      H(N) = H(N) + DT(N)*CAP(N)
       W(N) = W(N) + DT(N)*CAP(N)
      F(N)=F(N)+DFFF(N)
       DEX = DT(N)/DELTSX
      IF(NTYPE(N)) 3650,302,3650
 3650 IF(KCYC - 1) 3660, 3660,302
 3660 IF(NTYPE(N) - 2) 3670,310, 3670
 3670 DEX = DEX/(1.0 + DELTSX/SLIM(N))
  302 IF(DEX*DDT(N)) 3680,305,305
 3680 DEX = DEX*1.0E-24
  305 DDT(N) = DEX
  310 CONTINUE
      GENS = GENS + GS*DELTS
CCC   FIND NEW REACTANT OR PHASE CONCENTRATIONS.
      IF(NREACT)370,370, 3690
```

```
 3690 DO 320 N = 1,NODES
      A(N) = A(N) + DA(N)
      B(N) = B(N) + DB(N)
      K=NODMAT(N)
      HMX=HEFT(N)*HMELT(K)*DA(N)
      W(N)=W(N)-HMX
      H(N)=H(N)-HMX
      DEXA = DA(N)/DELTSX
       DEXB = DB(N)/DELTSX
      IF(NTYPE(N)) 3700,315, 3700
 3700  IF(KCYC - 1)320,320, 3710
 3710 CONTINUE
  315 IF(DEXA*DDA(N)) 3720,316,316
 3720 DEXA = DEXA*1.0E-24
  316 DDA(N) = DEXA
      IF(DEXB*DDB(N)) 3730,318,318
 3730 DEXB = DEXB*1.0E-24
  318 DDB(N) = DEXB
  320 CONTINUE
CCC   TEST CRITERIA FOR ENDING PROBLEM.
  370 IF(TMIN1 - TMIN) 3740,375,375
 3740  KWIT = 3
       GO TO 410
  375 IF(TMAX1 - TMAX)376,376, 3750
 3750  KWIT = 2
       GO TO 410
  376 IF(KSPEC)380, 3760, 3760
 3760 IF(DELTMX - BIG) 3770,378,378
 3770 IF(DELTMX - DELTO) 3780,378,378
 3780 IF(DELTMX - DELT) 3790, 3790,380
 3790 CONTINUE
  378 IF(RAT1-1.E3)380,3800,3800
 3800 IF(RAT2-1.E3)380,3810,3810
 3810 IF(KCYC-10)380,3820,3820
 3820 IF(KGOOD-3)380,3830,3830
 3830 KWIT=4
      GO TO 410
  380 IF(SUMTIM+DELT-TIMAX)385,385,3840
 3840 IF(SUMTIM-TIMAX)384,3850,3850
 3850 KWIT=1
      GO TO 410
  384 DELT=TIMAX-SUMTIM+1.0E-12
  385 IF(KCYC - MCYC)390, 3860, 3860
 3860  KWIT = 7
      GO TO 410
  390 IF(MOD(KCYC,2000/NODES))400,3870,400
 3870 CALL CLOCK1(KSEC,CLICKA)
      IF(MOD(86400+KSEC-KSECS,86400)-MSEC)400,3880,3880
```

```
 3880 KWIT=8                                                                16400
CCC   FIND OUT IF A PRINOUT IS REQUIRED NOW.                                16410
  400 IF(KWIT)700,3890,410                                                  16420
 3890 IF((KCYC-1)*MOD(KCYC,IPRINT))3900,3900,700                            16430
 3900 CONTINUE                                                              16440
  410 NPRINT=NPRINT+1                                                       16450
      NOW=1                                                                 16460
      HEAT=0.                                                               16470
      FLUX=0.                                                               16480
      TX=0.                                                                 16490
      CNN=0.0                                                               16500
      DO 414 K=1,NODES                                                      16510
      TMM=F(K)                                                              16520
  414 CNN=CNN+TMM                                                           16530
      FLUX=CNN                                                              16540
      DO 415 N=1,NODES                                                      16550
      HEAT=HEAT+W(N)                                                        16560
      TX=TX+CAP(N)*T(N)                                                     16570
  415 CONTINUE                                                              16580
      TEMPAD=TX/CAPS                                                        16590
      TEMPER=FLUX/CAPS                                                      16600
      TEMPLE=GENS/CAPS                                                      16610
      FX=FLUX/AMAX1(SUMTIM-TAU,1.0E-12)                                     16620
      TX = FX/CAPS                                                          16630
      WRITE(6,812)NPROB,NAME,CLOCKB,CLOCKA                                  16640
      WRITE(6,890)                                                          16650
      WRITE(6,815)NPRINT,KCYC,MF,MS,KWIT,DELTMX,
     1SMALL,TVARY,NUTS                                                      16670
      WRITE(6,820)SUMTIM,DELTS,FLUX,TEMPER,                                 16680
     1FX,TX,TEMPAD,CAPS,HEAT,GS,GENS,TEMPLE                                 16690
      WRITE(6,890)                                                          16700
      IF(KWIT)418,418,3910                                                  16710
 3910 KDATA=1                                                               16720
  418 IF(KCYC-1)420,420,3920                                                16730
 3920 IF(KDATA)3930,420,420
 3930 WRITE(6,875)(NODE(N),T(N),N=1,NODES)                                  16750
      WRITE(6,890)                                                          16760
      GO TO 500                                                             16770
  420 WRITE(6,885)                                                          16780
      DO 424 N=1,NODES                                                      16790
      IF(MOD(N+13,58))422,3940,422                                          16800
 3940 WRITE(6,885)                                                          16810
  422 WRITE(6,886)NODE(N),T(N),DT(N),DDT(N),G(N),W(N),H(N),F(N)             16820
     1,EQCURE(N)                                                            16830
  424 CONTINUE                                                              16840
      WRITE(6,890)                                                          16850
 3945 IF(NREACT)500,500,3950
 3950 NX=0                                                                  16870
```

```
      DO 430 N=1,NODES                                                  16880
       IF(A(N))425, 3960,425                                            16890
 3960  IF(B(N)) 3970,430, 3970                                          16900
 3970 CONTINUE                                                          16910
  425 NX = NX + 1                                                       16920
      IF(MOD(NX-1,28))428,3980,428
 3930 WRITE(6,880)                                                      16940
  428 WRITE(6,882)NODE(N),T(N),A(N),DA(N),DDA(N),                       16950
     $ B(N),DB(N),DDB(N)                                                16960
  430 CONTINUE                                                          16970
      WRITE(6,890)                                                      16980
  500 IF(KWIT)700,700,3990                                              16990
 3990 CONTINUE
      ENTRY TALLY1
      IF(NPUNCH) 4000,700, 4000                                         17010
 4000 CONTINUE                                                          17020
C PUNCH FINAL CONDITIONS IN BLOCK 9 FORMAT.                             17030
C     PUNCH STATEMENTS ARE CHANGED TO COMMENT.SEPT 1971
C     PUNCH 850,(NAME(I), I = 1,6),CLOCKB,CLOCKA                        17040
C     PUNCH 830, SUMTIM, NODES                                          17050
C     PUNCH 840,(NODE(N),T(N),A(N),B(N),GG(N), N = 1,NODES)             17060
      N = 0                                                             17070
      PUNCH 835,N                                                       17080
  700 IF(NOW) 4010, 4010,705                                            17090
 4010 IF(NUM)705,705, 4020                                              17100
 4020 N = NUM                                                           17110
      WRITE(6,955)NUMX,T(N),DDT(N),A(N),B(N),G(N),EQCURE(N),SUMTIM      17120
  705 IF(NDOT) 4030,750, 4030                                           17130
 4030 CONTINUE                                                          17140
  710 DO 720 N = 1,NODES                                                17150
      DDT(N)=0.0                                                        17160
      DDB(N)=0.0                                                        17170
      DDA(N)=0.0                                                        17180
  720 CONTINUE                                                          17190
  750 RETURN                                                            17200
C COMPLETED TALLY.  RETURN TO HEART.                                    17210
      END                                                               17220


      SUBROUTINE CHEM                                                   17230
C     CARDS COLUMN        CHEM VERSION 5/29/68.                         17240
C     FORTRAN             CHEM    TRUMP CHEMICAL REACTION SUB. FORTRAN-400   17250
CCC   VERSION 5/29/68.                                                  17260
      REAL*8 HMELT,A                                                    17270
      REAL*8 CON,DA,ZIP,TRAN                                            17280
      REAL*8 NAME,CAPS                                                  17290
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                              17300
      REAL*8 FOR,SLIM,T,GEOM                                            17310
```

```
    REAL*8 FORD                                                           17320
    COMMON NAME(20)
    COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
   1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
   2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
   3       T(355),ZIP(355),TRAN(950)
    COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
    COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
   1       ITEMS(15),NEWBL(15)
    COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
   1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
   2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
    COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
    COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
   1       DSTAR,DTMAX,FONE,GONE,HONE,PONE,
   2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
   3       TMAX,TMIN,TONE,TVARY
    COMMON HMELTX(15),LTABK(15),
   1       NLOOK(355,8),NODE(355),
   2       NODMAT(355),NTYPE(355),RADIUS(355),
   3       VOL(355),W(355),
   4       DFI(950),FI(950),NOD1(950),NOD2(950),
   5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
    COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
   1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
   2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
   3 NOXMAT(355),
   4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
    COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
   1       NOXE(355),NTYPES(355),TT(355)
    COMMON /ACHEMS/ KA(15),KAX(15),KB(15),KBX(15),
   1       B(355),BB(355),DB(355),DCB(355)
    COMMON /ACHEM/ AKEM(5),ET(12,5),KEM(5),LTABE(5),LTABQ(5),
   1       LTABZ(5),QT(12,5),SLOE(12,5),SLOQ(12,5),
   2       SLOZ(12,5),TVARE(12,5),TVARQ(12,5),
   3       TVARZ(12,5),ZT(12,5),
   4       EA(355),EB(355),QA(355),QB(355),ZA(355),ZB(355)
810 FORMAT (8E10.3)                                                       17650
815 FORMAT(10X,100(1H=))                                                  17660
820 FORMAT(10X,13HREACTANT DATA,//,11X,4HNODE,4X,6HA    B,4X,4HMASS,      17670
   1   8X,2HQA,10X,2HZA,10X,2HEA,10X,2HQB,10X,2HZB,10X,2HEB)              17680
825 FORMAT(10X,3I5,1X,1P7E12.4)                                           17690
877 FORMAT(/20X2HQT12X5HSLOPE10X5HTVARQ/,(15X,1P3E15.6))                  17700
878 FORMAT(/20X2HZT12X5HSLOPE10X5HTVARZ/,(15X,1P3E15.6))                  17710
879 FORMAT(/20X2HET12X5HSLOPE10X5HTVARE/,(15X,1P3E15.6))                  17720
899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)         17730
915 FORMAT(A5,4I5,5X,5E10.3)                                              17740
916 FORMAT(//,10X,36HNAME   REACT INDEX LTABQ LTABZ LTABE,8X,2HQT,11X,    17750
```

```
     1   2HZT,11X,2HET)
   917 FORMAT(10X,A5,I7,4I6,3X,1P3E13.5)
       IF(KCYC) 3000,100,200
  3000 CONTINUE
CARD BLOCK 3.  REACTANT PROPERTIES, VS TIME OR TEMPERATURE.
       N = NKEM
       LABEL = 1
       IF(MOE)10, 3010,10
  3010 NVARE=0
       NVARZ=0
       NVARQ=0
       NKEM=0
       N=0
    10 READ(5,915)A1,N1,L1,L2,L3,P1,P2,P3
       IF(N1) 3020,99, 3020
  3020 IF(MOE) 3030,20,20
  3030 CALL SEEK1(N,N1,KEM,NKEM,K)
       IF(K)18,18, 3040
  3040 IF(LTABQ(N)) 3050,14, 3050
  3050 NVARQ = NVARQ - 1
    14 IF(LTABZ(N)) 3060,16, 3060
  3060 NVARZ = NVARZ - 1
    16 IF(LTABE(N)) 3070,18, 3070
  3070 NVARE = NVARE - 1
    18 ITEMS(3)=MINO(N-1,ITEMS(3))
       GO TO 25
    20 NKEM=N+1
       N=N+1
    25 IF(N - M3)30,30, 3080
  3080 NKEM=M3
       N=M3
       KWIT = 11
       WRITE(6,899)N, IBLOCK
    30 AKEM(N)=A1
       KEM(N) = N1
       LTABQ(N) = L1
       LTABZ(N) = L2
       LTABE(N) = L3
       QT(1,N) = P1
       ZT(1,N) = P2
       ET(1,N) = P3
       IF(LABEL)76,76, 3090
  3090 LABEL = 0
       WRITE(6,916)
    76 WRITE(6,917)AKEM(N),KEM(N),N,LTABQ(N),LTABZ(N),
     $ LTABE(N), QT(1,N),ZT(1,N),ET(1,N)
       LTAB=IABS(LTABQ(N))
       IF(LTAB - 2) 3100, 73, 78
```

```
3100 LTABQ(N) = 0
      GO TO 85
  78 NVARQ = NVARQ + 1
      LABEL = 1
     READ(5,810)(QT(J,N),TVARQ(J,N),J=1,LTAB)
     DO  80 J=2,LTAB
     SLOQ(J,N) = (QT(J,N)-QT(J-1,N))/(TVARQ(J,N)-TVARQ(J-1,N))
  80 CONTINUE
     SLOQ(1,N) = SLOQ(2,N)
     WRITE(6,877)(QT(J,N),SLOQ(J,N),TVARQ(J,N),J=1,LTAB)
     IF(LTAB - M9)85,85, 3110
3110 KWIT = 12
  85 LTAB=IABS(LTABZ(N))
      IF(LTAB - 2) 3120, 88, 88
3120  LTABZ(N) = 0
      GO TO 95
  88 NVARZ = NVARZ + 1
      LABEL = 1
     READ(5,810)(ZT(J,N),TVARZ(J,N),J=1,LTAB)
     DO 90 J=2,LTAB
     SLOZ(J,N) = (ZT(J,N)-ZT(J-1,N))/(TVARZ(J,N)-TVARZ(J-1,N))
  90 CONTINUE
     SLOZ(1,N) = SLOZ(2,N)
     WRITE(6,878)(ZT(J,N),SLOZ(J,N),TVARQ(J,N),J=1,LTAB)
     IF(LTAB - M9)95,95, 3130
3130 KWIT = 12
  95 LTAB=IABS(LTABE(N))
     IF(LTAB - 2) 3140,96,96
3140 LTABE(N) = 0
     GO TO 10
  96 NVARE = NVARE + 1
      LABEL = 1
     READ(5,810)(ET(J,N),TVARE(J,N),J=1,LTAB)
     DO 97 J=2,LTAB
     SLOE(J,N) = (ET(J,N)-ET(J-1,N))/(TVARE(J,N)-TVARE(J-1,N))
  97 CONTINUE
     SLOE(1,N) = SLOE(2,N)
     WRITE(6,879)(ET(J,N),SLOE(J,N),TVARE(J,N),J=1,LTAB)
     IF(LTAB - M9)10,10, 3150
3150 KWIT = 12
     GO TO 10
  99 NEWBL(2) = NEWBL(2) + 1000
     RETURN
C COMPLETED BLOCK 3.
 100 IBLOCK = 2
     CALL  REFER(KA,KAX,NMAT,KEM,NKEM)
     CALL  REFER(KB,KBX,NMAT,KEM,NKEM)
     IF(KWIT)600, 3160,600
```

```
 3160 RGAS = 1.987
CCC   FIND NODE REACTION HEATS, COLL.FREQ., AND ACTIVATION ENERGIES.
      DO 150 N = 1,NODES
       QA(N) = 0.
       QB(N) = 0.
      ZA(N) = 0.
      ZB(N) = 0.
      EA(N) = 0.
      EB(N) = 0.
      L1=NODMAT(N)
      LA=KA(L1)
      IF(LA)140,140, 3170
 3170 QA(N) = QT(1,LA)
      ZA(N) = ZT(1,LA)
      EA(N) = ET(1,LA)
  140 L1=NODMAT(N)
      LB=KB(L1)
      IF(LB)150,150, 3180
 3180 QB(N) = QT(1,LB)
      ZB(N) = ZT(1,LB)
      EB(N) = ET(1,LB)
  150 CONTINUE
  200 FORD = FOR*DELT
      IF(NOW)208,208, 3190
 3190 IF(KCYC - 1)208,205, 3200
 3200 IF(KDATA)208,208, 3210
 3210  IF(NVARZ + NVARQ + NVARE)208,208, 3220
 3220 CONTINUE
  205 WRITE(6,820)
      DO 206 N=1,NODES
      J = NODMAT(N)
      IF(KA(J))206,206, 3230
 3230 WRITE(6,825)NODE(N),KAX(J),KBX(J),HEFT(N),
     1   QA(N),ZA(N),EA(N),QB(N),ZB(N),EB(N)
  206 CONTINUE
      WRITE(6,815)
  208 IF(NVARQ)300,300, 3240
 3240 CONTINUE
CCC   FIND NEW HEATS OF REACTION.
      DO 290 N= 1,NODES
      L1=NODMAT(N)
      IF(KA(L1))3250,210,3250
 3250  NN = 1
      K=KA(L1)
       LTA3 = LTABQ(K)
       BETA = QA(N)
      GO TO 215
  210 L1=NODMAT(N)
```

```
      IF(KB(L1))3260,290,3260
 3260  NN = 2
      K=KB(L1)
       LTAB = LTABQ(K)
       BETB = QB(N)
  215 IF(LTAB) 3270,285,220
 3270  SET = SUMTIM + FORD
      GO TO 230
  220 L = NLOOK(N,3)
       SET = T(L) + FORD*DDT(L)
  230 MIN = 1
      MAX=IABS(LTAB)
  240 MID = (MIN + MAX)/2
       IF(SET - TVARQ(MID,K))250,270,260
  250 MAX = MID
       IF(MAX -2)270,240,240
  260 MIN = MID
       IF(MAX - MIN - 2)270,240,240
  270 SET = QT(MID,K) + SLOQ(MID+1,K)*(SET - TVARQ(MID,K))
      GO TO (275,280),NN
  275 QA(N) = SET
      BETA=100.0*ABS((BETA-QA(N))/(1.0E-24+BETA))
      DTMAX=AMAX1(DTMAX,BETA*TVARY)
       GO TO 210
  280 QB(N) = SET
      BETB=100.0*ABS((BETB-QB(N))/(1.0E-24+BETB))
      DTMAX=AMAX1(DTMAX,BETB*TVARY)
  285 GO TO (210,290),NN
  290 CONTINUE
CCC   FIND NEW COLLISION FREQUENCIES.
  300 IF(NVARZ)400,400, 3280
 3280 DO 390 N = 1,NODES
      L1=NODMAT(N)
      IF(KA(L1))3290,310,3290
 3290  NN = 1
      K=KA(L1)
      LTAB = LTABZ(K)
      BETA = ZA(N)
      GO TO 315
  310 L1=NODMAT(N)
      IF(KB(L1))3300,390,3300
 3300  NN = 2
      K=KB(L1)
       LTAB = LTABZ(K)
      BETB = ZB(N)
  315 IF(LTAB) 3310,385,320
 3310  SET = SUMTIM + FORD
      GO TO 330
```

```
  320 L = NLOOK(N,4)
      SET = T(L) + FORD*DDT(L)
  330 MIN = 1
      MAX=IABS(LTAB)
  340 MID = (MIN + MAX)/2
       IF(SET - TVARZ(MID,K))350,370,360
  350 MAX = MID
       IF(MAX - 2)370,340,340
  360 MIN = MID
       IF(MAX - MIN - 2)370,340,340
  370 SET = (ZT(MID,K) + SLOZ(MID+1,K)*(SET - TVARZ(MID,K)))
      GO TO (375,380),NN
  375 ZA(N) = SET
      BETA=100.0*ABS((BETA-ZA(N))/(1.0E-24+BETA))
      DTMAX=AMAX1(DTMAX,BETA*TVARY)
       GO TO 310
  380 ZB(N) = SET
      BETB=100.0*ABS((BETB-ZB(N))/(1.0E-24+BETB))
      DTMAX=AMAX1(DTMAX,BETB*TVARY)
  385 GO TO (310,390),NN
  390 CONTINUE
CCC   FIND NEW ACTIVATION ENERGIES.
  400 IF(NVARE)500,500, 3320
 3320 DO 490 N= 1,NODES
      L1=NODMAT(N)
      IF(KA(L1))3330,410,3330
 3330  NN = 1
      K=KA(L1)
       LTAB = LTABE(K)
       BETA = EA(N)
       GO TO 415
  410 IF(KB(L1))3340,490,3340
 3340  NN = 2
      K=KB(L1)
       LTAB = LTABE(K)
      BETB = EB(N)
  415 IF(LTAB) 3350,485,420
 3350  SET = SUMTIM + FORD
       GO TO 430
  420 L=NLOOK(N,5)
       SET = T(L) + FORD*DDT(L)
  430 MIN = 1
      MAX=IABS(LTAB)
  440 MID = (MIN + MAX)/2
       IF(SET - TVARE(MID,K))450,470,460
  450 MAX = MID
       IF(MAX - 2)470,440,440
  460 MIN = MID
```

```
       IF(MAX - MIN - 2)470,440,440                                             00130
  470 SET = (ET(MID,K) + SLOE(MID+1,K)*(SET - TVARE(MID,K)))                    00140
      GO TO (475,480),NN                                                        00150
  475 EA(N) = SET                                                               00160
      BETA=100.0*ABS((BETA-EA(N))/(1.0E-24+BETA))                               00170
      DTMAX=AMAX1(DTMAX,BETA*TVARY)                                             00180
       GO TO 410                                                                00190
  480 EB(N) = SET                                                               00200
      BETB=100.0*ABS((BETB-EB(N))/(1.0E-24+BETB))                               00210
      DTMAX=AMAX1(DTMAX,BETB*TVARY)                                             00220
  485 GO TO (410,490),NN                                                        00230
  490 CONTINUE                                                                  00240
CCC   FIND CONCENTRATION AND TEMP CHANGE IN NODES.                              00250
  500 DO 520 N = 1,NODES                                                        00260
       TEX = (T(N) + FORD*DDT(N) + TBASE)*RGAS                                  00270
      L1=NODMAT(N)                                                              00280
      IF(KA(L1))3360,510,3360                                                   00290
 3360 DEXA=DELT*EXP(AMIN1(50.0,ZA(N)-EA(N)/TEX))                                00300
      DEXA=-A(N)*AMIN1(1.0,DEXA)                                                00310
      DT(N) = DT(N) - QA(N)*DEXA*HEFT(N)/CAP(N)                                 00320
      DA(N) = DA(N) + DEXA                                                      00330
  510 K1=NODMAT(N)                                                              00340
      IF(KB(K1))3370,520,3370                                                   00350
 3370 DEXB=DELT*EXP(AMIN1(50.0,ZB(N)-EB(N)/TEX))                                00360
      DEXB=-B(N)*AMIN1(1.0,DEXB)                                                00370
      DT(N) = DT(N) - QB(N)*DEXB*HEFT(N)/CAP(N)                                 00380
      DB(N) = DB(N) + DEXB                                                      00390
  520 CONTINUE                                                                  00400
  600 RETURN                                                                    00410
CCC   END OF CHEM. GO BACK TO HEART.                                            00420
      END                                                                       00430

      SUBROUTINE SPECK                                                          00440
C     CARDS COLUMN       SPECK VERSION  5/29/68.                                00450
C     LIST 8                                                                    00460
C     FORTRAN            SPECK   TRUMP SPECIAL NODE CALC. SUB. FORTRAN400       00470
CCC   VERSION  5/29/68.                                                         00480
CCC   CONTAINS INSTRUCTIONS REQUIRED ONLY WITH FLOW (260, 365, 460).            00490
CCC   CONTAINS INSTRUCTIONS REQUIRED ONLY WITH SURE (650).                      00500
      REAL*8 HMELT,A                                                            00510
      REAL*8 EX                                                                 00520
      REAL*8 CON,DA,ZIP,TRAN                                                    00530
      REAL*8 TRANS                                                              00540
      REAL*8 NAME,CAPS                                                          00550
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                      00560
      REAL*8 FOR,SLIM,T,GEOM                                                    00570
      REAL*8 FLEX                                                               00580
```

```
      REAL*8 FORD                                                       00590
      REAL*8 DFS                                                        00600
      REAL*8 ERROR,ERRORX                                               00610
      REAL*8 FLOWN                                                      00620
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      COMMON /AFLOWS/ FLOWN(50),NODF1(50),NODF2(50)
      COMMON /ASPECK/ ERROR(355),ERRORX(355)
      COMMON /ASURES/ DFS(60),TRANS(60),NODS(60)
  815 FORMAT(/10X,6HKCYC =,I6,5X,35HNO CONVERGENCE.  WILL REPEAT CYCLE.)
  820 FORMAT(/10X,6HKCYC =,I6,5X,81HNO CONVERGENCE.  CAN NOT REPEAT CYCL   00920
     1E, SO WILL END PROBLEM.  YOU MAY FIX BY USING,/,27X,73HLARGER NODE   00930
     2S, LUMPING NODES, OR ELIMINATING HIGH CONDUCTANCE CONNECTIONS.)      00940
  825 FORMAT(10X,28HTOTAL NUMBER OF ITERATIONS =,I6,5X,5HMAX =,I6)
      IF(KCYC) 3000,200,200                                             00960
 3000  NUTX = 0                                                         00970
       NUTSUM = 0                                                       00980
       NUTMAX = 200                                                     00990
CCC   INITIALIZE CORRECTIONS TO TEMP CHANGES IN NODES.                  01000
      DO 120 N = 1,NODES                                                01010
      ERRORX(N) = 0.0                                                   01020
```

```
  120 CONTINUE
CCC    SET ACCELERATION FACTOR AND CONVERGENCE FACTOR.
       NUTS = 0
        SPEED = 0.2
        ERRS = 5.E-5
        RETURN
  200 IF(KWIT)210,210, 3010
 3010 CONTINUE
C STRART HERE IF KCYC ) -1.
      WPITE(6,825)NUTSUM,NUTX
      GO TO 700
  210 FORD = FOR*DELT
CCC    INITIZLIZE CORRECTION FACTORS.
      DO 220 N = 1,NODES
      ERROR(N) = 0.0
      IF(NTYPE(N)) 3020,220, 3020
 3020  ERRORX(N) = DELT*DDT(N)
  220 CONTINUE
CCC    CALC CORRECTION FACTORS FOR SPECIAL NODES, COUNT SPEC-SPEC CONN.
      KNOT=0
      KNOFL=0
      KNOCK=0
      IF(NOCON)260,260, 3030
 3030 DO 250 N = 1,NOCON
      N1 = NOD1(N)
      N2 = NOD2(N)
      IF(NTYPE(N1))230, 3040,230
 3040 IF(NTYPE(N2)) 3050,250, 3050
 3050 ERROR(N2) = ERROR(N2) + TRAN(N)*DT(N1)
      GO TO 250
  230 IF(NTYPE(N2))240, 3060,240
 3060 ERROR(N1) = ERROR(N1) + TRAN(N)*DT(N2)
      GO TO 250
  240 ERROR(N1) = ERROR(N1) + TRAN(N)*ERRORX(N2)
      ERROR(N2) = ERROR(N2) + TRAN(N)*ERRORX(N1)
       KNOCK = KNOCK + 1
  250 CONTINUE
  260 CONTINUE
C REMOVE INSTRUCTIONS PRECEDING 300 IF FLOW NOT USED.
      IF(NOFLOW)300,300, 3070
 3070 CONTINUE
CCC    FIND CORRECTIONS DUE TO MASS FLOW.
      DO 280 N = 1,NOFLOW
      N1 = NODF1(N)
      N2 = NODF2(N)
      IF(NTYPE(N1))270, 3080,270
 3080 IF(NTYPE(N2)) 3090,280, 3090
 3090 ERROP(N2) = ERROR(N2) + FLOWN(N)*CAP(N1)*DT(N1)/HEFT(N1)
```

```
      GO TO 280                                                         01510
  270 IF(NTYPE(N2)) 3100,280, 3100                                      01520
 3100 KNOFL = KNOFL + 1                                                 01530
      ERROR(N2)=ERROR(N2) + FLOWN(N)*CAP(N1)*ERRORX(N1)/HEFT(N1)        01540
  280 CONTINUE                                                          01550
CCC   MAKE FIRST CORRECTION TO TEMP CHANGES IN SPECIAL NODES.           01560
  300 NUTS = 0                                                          01570
       ESUM = 0.0                                                       01580
      HSUM = 1.E-12                                                     01590
       S1 = SPEED                                                       01600
       S2 = S1 + 1.                                                     01610
      IF((KNOCK + KNOFL)*KCYC) 3110, 3110,310                           01620
 3110 S1 = 0.0                                                          01630
      S2 = 1.0                                                          01640
  310 DO 320 N = 1,NODES                                                01650
      IF(NTYPE(N)) 3120,320, 3120                                       01660
 3120 EX = ERROR(N) + S1*ZIP(N)*ERRORX(N)                               01670
      ERROR(N) = 0.                                                     01680
      DT(N) = (CAP(N)*DT(N) + FORD*EX)/(CAP(N) + FORD*S2*ZIP(N))        01690
      ERRORX(N) = DT(N)-ERRORX(N)                                       01700
      ESUM=ESUM+CAP(N)*DABS(ERRORX(N))                                  01710
      HSUM = HSUM + CAP(N)                                              01720
CCC   CAPACITY OF SPECIAL NODES.                                        01730
  320 CONTINUE                                                          01740
       HSUM = HSUM*TVARY                                                01750
C MAX CHANGE IN HEAT CONTENT.                                           01760
CCC   START ITERATING IF THERE ARE SPEC-SPEC CONNECTIONS, AND           01770
CCC   RELATIVE CHANGE IN HEAT CONTENT IS GREATER THAN ERRS.             01780
      IF(ESUM - ERRS*HSUM)400, 3130, 3130                               01790
 3130 IF(KNOCK + KNOFL)400,400, 3140                                    01800
 3140 CONTINUE                                                          01810
  340 NUTS = NUTS + 1                                                   01820
C CALC CORRECTIONS FOR SPECIAL NODES.                                   01830
      IF(KNOCK)365,365, 3150                                            01840
 3150 DO 360 N = 1,NOCON                                                01850
      N1 = NOD1(N)                                                      01860
      N2 = NOD2(N)                                                      01870
      IF(NTYPE(N1)*NTYPE(N2)) 3160,360, 3160                            01880
 3160 CONTINUE                                                          01890
CCC   BOTH MUST BE SPECIAL.                                             01900
      ERROR(N1) = ERROR(N1) + TRAN(N)*ERRORX(N2)                        01910
      ERROR(N2) = ERROR(N2) + TRAN(N)*ERRORX(N1)                        01920
  360 CONTINUE
  365 CONTINUE                                                          01940
C REMOVE INSTRUCTIONS PRECEDING 375 IF FLOW NOT USED.                   01950
      IF(KNOFL)375,375, 3170                                            01960
 3170 CONTINUE                                                          01970
CCC   FIND CORRECTIONS DUE TO MASS FLOW.                                01980
```

```
      DO 370 N = 1,NOFLOW                                                 01990
      N1 = NODF1(N)                                                       02000
      N2 = NODF2(N)                                                       02010
      IF(NTYPE(N1)*NTYPE(N2)) 3180,370, 3180                              02020
 3180 ERROR(N2)=ERROR(N2) + FLOWN(N)*CAP(N1)*ERRORX(N1)/HEFT(N1)          02030
  370 CONTINUE
CCC CORRECT TEMP CHANGE IN SPECIAL NODES.                                 02050
  375 ESUM = 0.                                                           02060
      DO 380 N = 1,NODES                                                  02070
      IF(NTYPE(N)) 3190,380, 3190                                         02080
 3190 ERRORX(N) = (ERROR(N)+S1*ZIP(N)*ERRORX(N))/(S2*ZIP(N)+CAP(N)/FORD)  02090
      DT(N) = DT(N) + ERRORX(N)                                           02100
       ERROR(N) = 0.                                                      02110
      ESUM=ESUM+CAP(N)*DABS(ERRORX(N))                                    02120
  380 CONTINUE                                                            02130
CCC   STOP ITERATING AFTER NUTMAX CYCLES.                                 02140
      IF(NUTS - NUTMAX)390, 3200, 3200                                    02150
 3200 NUTS=0                                                              02160
      DTMAX=AMAX1(DTMAX,200.0*TVARY)                                      02170
      WRITE(6,815)KCYC                                                    02180
      IF(DELT - 2.0*SMALL) 3210, 3210,400                                 02190
 3210 KWIT = 10                                                           02200
C     ITERATION FAILURE                                                   02210
      WRITE(6,820)KCYC                                                    02220
      GO TO 700                                                           02230
CCC   STOP ITERATING WHEN RELATIVE ERROR IN HEAT CONTENT CHANGES ERRS     02240
  390 IF(ESUM-ERRS*HSUM) 3220,3220,340                                    02250
 3220 CONTINUE                                                            02260
  400 NUTSUM=NUTSUM+NUTS                                                  02270
      NUTX=MAX0(NUTX,NUTS)                                                02280
CCC   CORRECT INTERVAL FLUXES AND FIND CORRECTION FACTORS FOR REG NODES   02290
      IF(NOCON)460,460,3230                                               02300
 3230 DO 450 N=1,NOCON                                                    02310
      N1=NOD1(N)                                                          02320
      N2=NOD2(N)                                                          02330
      IF(NTYPE(N1))420,3240,420                                           02340
 3240 IF(NTYPE(N2))3250,450,3250                                          02350
 3250 ERROR(N1)=ERROR(N1)+TRAN(N)*(DT(N2)-DT(N1))                         02360
      GO TO 440                                                           02370
  420 IF(NTYPE(N2))440, 3260,440
 3260 ERROR(N2)=ERROR(N2)+TRAN(N)*(DT(N1)-DT(N2))                         02390
  440 FLEX=FORD*TRAN(N)*(DT(N2)-DT(N1))                                   02400
      DFI(N)=DFI(N)+FLEX                                                  02410
      DF(N1)=DF(N1)+FLEX                                                  02420
      DF(N2)=DF(N2)-FLEX                                                  02430
  450 CONTINUE                                                            02440
  460 CONTINUE                                                            02450
C     REMOVE INSTRUCTIONS PRECEDING 500 IF FLOW NOT USED                  02460
```

```
      IF(NOFLOW)500,500,3270                                            02470
 3270 DO 480 N=1,NOFLOW                                                 02480
      N1=NODF1(N)                                                       02490
      N2=NODF2(N)                                                       02500
      IF(NTYPE(N1)) 480,3280,480                                        02510
 3280 IF(NTYPE(N2)) 3290,480,3290                                       02520
 3290 ERROR(N1)=ERROR(N1)-FLOWN(N)*CAP(N1)*DT(N1)/HEFT(N1)              02530
  480 CONTINUE                                                          02540
CCC   CALC CORRECTED TEMP CHANGE IN REG NODES CONN TO SPECIAL NODES.    02550
  500 DO 520 N=1,NODES                                                  02560
      IF(NTYPE(N))520,3300,520                                          02570
 3300 DT(N) = DT(N) + FORD*ERROR(N)/CAP(N)                              02580
  520 CONTINUE                                                          02590
  650 CONTINUE                                                          02600
C     REMOVE INSTRUCTIONS PRECEDING 700 IF SURE NOT USED                02610
      IF(NOSCON)700,700,3310                                            02620
 3310 CONTINUE                                                          02630
CCC   CORRECT FLUX OF EXTERNAL CONNECTIONS                              02640
      DO 660 N=1,NOSCON                                                 02650
      N1=NODS(N)                                                        02660
      FLEX=FORD*TRANS(N)*DT(N1)                                         02670
      DF(N1)=DF(N1)-FLEX                                                02680
      DFS(N)=DFS(N)-FLEX                                                02690
  660 CONTINUE                                                          02700
  700 RETURN                                                            02710
C     COMPLETED SPECK. RETURN TO HEART.                                 02720
      END                                                               02730


      SUBROUTINE FINK                                                   02740
C     CARDS COLUMN     FINK VERSION 5/29/68.                            02750
C     LIST 8                                                            02760
C     FORTRAN          FINK TRUMP  INTERNAL HEAT FLOW SUB.FORTRAN-400   02770
      REAL*8 HMELT,A                                                    02780
      REAL*8 AREA,DEL1,DEL2                                             02790
      REAL*8 CON,DA,ZIP,TRAN                                            02800
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                              02810
      REAL*8 FOR,SLIM,T,GEOM                                            02820
      REAL*8 FORD                                                       02830
      REAL*8 NAME,CAPS                                                  02840
      REAL*8 HEX                                                        02850
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
```

```
     1          ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1          NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2          NVARZ,NVARE,NORAD,NVARH,NORADS,NOPCWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1          DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2          RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3          TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1          NLOOK(355,8),NODE(355),
     2          NODMAT(355),NTYPE(355),RADIUS(355),
     3          VOL(355),W(355),
     4          DFI(950),FI(950),NOD1(950),NOD2(950),
     5          F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1          LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2          TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4          NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1          NOXF(355),NTYPES(355),TT(355)
      COMMON /AFINK/ AREA(950),DEL1(950),DEL2(950),HINT(950),
     1          NOX1(950),NOX2(950),RINT(950)
  818 FORMAT(10X,I5,I6,I5,1P7E12.4)
  820 FORMAT(2I5,3I3,I1,4E10.3,20A1)
  821 FORMAT(//,11X,93HNOD1 NOD2 INDEX  DEL1          DEL2          DLONG
     $     DRAD          HINT          RINT          AREA)
  825 FORMAT(10X,100(1H=))
  830 FORMAT(10X,24HINTERNAL CONNECTION DATA)
  835 FORMAT(/,10X,87H      NOD1  NOD2     AREA          HINT          RINT
     $      TRAN          HEAT FLOW    AVG RATE)
  840 FORMAT(13X,2I6,1X,6F12.4)
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)
      IF(KCYC) 3000,100,200
 3000 CONTINUE
CARD BLOCK 5.  INTERNAL CONNECTION NODE NUMBERS, DESCRIPTIONS.
      N = NOCON
      L = 0
      IF(MOE)2, 3010,2
 3010 NCRAD=0
      NOCON=0
      N=0
    2 READ(5,820)N1,N2,NSEQ,NAD1,NAD2,NZ,P1,P2,
     1    DLONG,DRAD,NX1,NX2
      IF(N1) 3020,30, 3020
 3020 IF(NZ)4,4, 3030
 3030 NZ = 10**NZ
```

```
      NAD1 = NAD1*NZ                                                    03370
      NAD2 = NAD2*NZ                                                    03380
    4 CALL PATCH(NX1,1.E12,HINTS,LBH)                                   03390
      CALL PATCH(NX2,0.0,RINTS,LBR)                                     03400
      IF(LBR) 3040,6, 3040                                              03410
 3040 NORAD = NORAD + 1 + NSEQ                                          03420
      IF(LBH) 3050,7, 3050                                              03430
 3050 CONTINUE                                                          03440
    6 IF(HINTS)8, 3060,8                                                03450
 3060 CONTINUE                                                          03460
    7 HINTS = 1.0E-24                                                   03470
    8 ADD = 0.0                                                         03430
      IF(DRAD) 3070,10,10                                               03490
 3070 ADD = -(DRAD + DRADS)                                             03500
      DRAD = DRADS                                                      03510
   10 IF(MOE) 3080,12,12
 3080 CALL SEEK2(N,N1,N2,NOX1,NOX2,NOCON,K)                             03530
      IF(K)11,11, 3090                                                  03540
 3090 IF(PINT(N))  3100,11, 3100                                        03550
 3100 NORAD = NORAD - 1                                                 03560
   11 ITEMS(5)=MINO(N-1,ITEMS(5))                                       03570
      GO TO 14                                                          03580
   12 NOCON=N+1                                                         03590
      N=N+1                                                             03600
   14 IF(N - M5)16,16, 3110
 3110 NOCON=M5                                                          03620
      N=M5                                                              03630
      KWIT= 11                                                          03640
      WRITE(6,899)N, IBLOCK                                             03650
   16 NOX1(N)=N1                                                        03660
      NOD1(N)=N1                                                        03670
      NOX2(N)=N2                                                        03680
      NOD2(N)=N2                                                        03690
      DEL1(N) = P1*SCALE                                                03700
      DEL2(N) = P2*SCALE                                                03710
      HINT(N) = HINTS                                                   03720
      RINT(N) = RINTS                                                   03730
      DRADS=DRAD+ADD                                                    03740
      DRAD=DRADS                                                        03750
      AREA(N) = GEOM*DLONG*DRAD**KSYM*SCALE**2                          03760
      L = L + 1                                                         03770
      IF(MOD(L-1,57))20,3120,20                                         03780
 3120 WRITE(6,821)                                                      03790
   20 WRITE(6,818)NOD1(N),NOD2(N),N,DEL1(N),DEL2(N),                    03800
     1 DLONG,DRAD,HINT(N),RINT(N),AREA(N)                               03810
      IF(NSEQ)2,2, 3130                                                 03820
 3130 NSEQ = NSEQ - 1                                                   03830
      N1 = N1 + NAD1                                                    03840
```

```
      N2 = N2 + NAD2
      GO TO 10
   30 RETURN
C COMPLETED BLOCK 5.
  100 IBLOCK = 5
      CALL REFER(NOD1,NOX1,NOCON,NODE,NODES)
      CALL REFER(NOD2,NOX2,NOCON,NODE,NODES)
      IF(KWIT)700, 3140,700
 3140 CONTINUE
CCC   CALCULATE CONDUCTANCE OF INTERNAL CONNECTIONS.
      DO 180 N = 1,NOCON
      N1 = NOD1(N)
      N2 = NOD2(N)
      T1 = TBASE + T(N1)
       T2 = TBASE + T(N2)
      DT12=AMAX1(1.0E-24,ABS(T1-T2))
      RINTR=AMAX1(0.0,RINT(N))*SIGMA
      RINTC=AMAX1(0.0,-RINT(N))
      RAD = RINTR  *(T1 + T2)*(T1*T1 + T2*T2)
      RAD = 1.0/(RAD + HINT(N)*DT12**RINTC)
      TRAN(N) = AREA(N)/(DEL1(N)/CON(N1) + DEL2(N)/CON(N2) + RAD)
      ZIP(N1) = ZIP(N1) + TRAN(N)
       ZIP(N2) = ZIP(N2) + TRAN(N)
      DFI(N) = 0.
      FI(N) = 0.
  180 CONTINUE
CCC   FIND TOTAL HEAT FLUX ACROSS EACH INTERNAL CONNECTION.
  200 DO 215 N = 1,NOCON
  215 FI(N)= FI(N) + DFI(N)
      IF(NOW)300,300, 3150
 3150  IF(KCYC - 1)300,225, 3160
 3160  IF(KDATA)300,300, 3170
 3170 CONTINUE
CCC   WRITE OUT PROPERTIES OF EACH INTERNAL CONNECTION.
  225 WRITE(6,830)
      TX=AMAX1(SUMTIM -TAU,1.0E-12)
      DO 240 N = 1,NOCON
      IF(MOD(N-1,57))235,3180,235
 3180 WRITE(6,835)
  235 FX = FI(N)/TX
      WRITE(6,840)NOX1(N),NOX2(N),AREA(N),
     $ HINT(N),RINT(N),TRAN(N),FI(N),FX
  240 CONTINUE
      WRITE(6,825)
CCC   FIND NEW CONDUCTANCES OF INTERNAL CONNECTIONS.
  300 IF(NVARK + NORAD)500,500, 3190
 3190  FORD = FOR*DELT
      DO 420 N = 1,NOCON
```

```
      N1 = NOD1(N)                                                                 04330
      N2 = NOD2(N)                                                                 04340
      IF(RINT(N))415, 3200,415                                                     04350
 3200  J1 = NODMAT(N1)                                                             04360
       J2 = NODMAT(N2)                                                             04370
      IF(LTABK(J1))415, 3210,415                                                   04380
 3210  IF(LTABK(J2)) 3220,420, 3220                                                04390
 3220 CONTINUE                                                                     04400
  415 ZIP(N1) = ZIP(N1) - TRAN(N)                                                  04410
       ZIP(N2) = ZIP(N2) - TRAN(N)                                                 04420
      T1 = TBASE + T(N1)+FORD*DDT(N1)                                              04430
      T2 = TBASE + T(N2)+FORD*DDT(N2)                                              04440
      DT12=AMAX1(1.0E-24,ABS(T1-T2))                                               04450
      RINTR=AMAX1(0.0,RINT(N))*SIGMA                                               04460
      RINTC=AMAX1(0.0,-RINT(N))                                                    04470
      RAD = RINTR  *(T1 + T2)*(T1*T1 + T2*T2)                                      04480
      RAD = 1.0/(RAD + HINT(N)*DT12**RINTC)                                        04490
      TRAN(N) = AREA(N)/(DEL1(N)/CON(N1) + DEL2(N)/CON(N2) + RAD)                  04500
      ZIP(N1) = ZIP(N1) + TRAN(N)                                                  04510
       ZIP(N2) = ZIP(N2) - TRAN(N)
  420 CONTINUE                                                                     04520
CCC   FIND TEMPERATURE CHANGES IN NODES DUE TO CONDUCTION.                         04530
  500 DO 510 N=1,NOCON                                                             04540
      N1=NOD1(N)                                                                   04550
      N2=NOD2(N)                                                                   04560
      HEX = DELT*TRAN(N)*(T(N2) - T(N1))                                           04570
       DFI(N) = HEX                                                                04580
      DF(N1) = DF(N1) + HEX                                                        04590
       DF(N2) = DF(N2) - HEX                                                       04600
      DT(N1) = DT(N1) + HEX/CAP(N1)                                                04610
       DT(N2) = DT(N2) - HEX/CAP(N2)                                               04620
  510 CONTINUE                                                                     04630
  700 RETURN                                                                       04640
C COMPLETED FINK. RETURN TO HEART.                                                 04650
      END


      SUBROUTINE GEN                                                               04670
CCC   VERSION 5/29/68                                                              04680
      REAL*8 CON,DA,ZIP,TRAN                                                       04690
      REAL*8 NAME,CAPS                                                             04700
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                         04710
      REAL*8 FORD                                                                  04720
      REAL*8 FOR,SLIM,T,GEOM                                                       04730
      REAL*8 HMELT,A                                                               04740
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
```

```
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      COMMON /AGEN/ GT(12,5),LTABG(5),NODG(5),NOXG(5),
     1       SLOG(12,5),TVARG(12,5)
  810 FORMAT (8E10.3)                                                    05020
  832 FORMAT(10X,3I6,3X,1PE15.6,15X,E15.6)                               05030
  833 FORMAT(31X,1P3E15.6)                                               05040
  870 FORMAT (4I5,6E10.3)                                                05050
  871 FORMAT(//,12X,16HNODG INDEX LTABG,8X,2HGT,12X,5HSLOPE,11X,5HTVARG) 05060
  872 FORMAT(/,10X,66HFOLLOWING NODE HAS G(N) = GT(1,N)*EXPF(-0.69315*SU 05070
     1MTIM/TVARG(1,N)))                                                  05080
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)      05090
      IF(KCYC) 3000,100,200                                              05100
 3000 CONTINUE                                                           05110
CARD BLOCK 8.  TABLES OF HEAT GENERATION VS TIME OR TEMPERATURE.         05120
      N = NVARG                                                          05130
      LABEL = 1                                                          05140
      IF(MOE)2, 3010,2                                                   05150
 3010 NVARG=0                                                            05160
      N=0                                                                05170
    2 READ(5,870)N1,NSEQ,NADG,L1,(F1(J),J=1,3)                           05180
      IF(N1) 3020,30, 3020                                               05190
 3020 LTAB=IABS(L1)                                                      05200
      IF(LTAB - 2) 3030,4,4                                              05210
```

```
3030 LTAB=0
     L1=0
     IF(F2(1))10, 3040,10
3040 F2(1) = 1.0E-24
     GO TO 10
   4 IF(LTAB - 4)7, 3050, 3050
3050 IF(LTAB - M9)6,6, 3060
3060 KWIT = 12
   6 READ(5,810)(F1(J),F2(J),J=4,LTAB)
   7 DO 8 J = 2,LTAB
     F3(J) = (F1(J) - F1(J-1))/(F2(J) - F2(J-1))
   8 CONTINUE
     F3(1) = F3(2)
  10 IF(MOE) 3070,12,12
3070 CALL SEEK1(N,N1,NOXG,NVARG,K)
     ITEMS(8)=MIN0(N-1,ITEMS(8))
     GO TO 14
  12 NVARG=N+1
     N=N+1
  14 IF(N - M8)16,16, 3080
3080 NVARG=M8
     N=M8
     KWIT = 11
     WRITE(6,899)N,IBLOCK
  16 NOXG(N)=N1
     NODG(N)=N1
     LTABG(N) = L1
     GT(1,N) = F1(1)
     TVARG(1,N) = F2(1)
     IF(L1)18, 3090,18
3090 LABEL = 1
     WRITE(6,872)
  18 IF(LABEL)20,20, 3100
3100 LABEL = 0
     WRITE(6,871)
  20 WRITE(6,832)NODG(N),N,LTABG(N),GT(1,N),TVARG(1,N)
     IF(LTAB)24,24,3110
3110 DO 22 J = 2,LTAB
     GT(J,N) = F1(J)
     TVARG(J,N) = F2(J)
     SLOG(J,N) = F3(J)
  22 CONTINUE
     SLOG(1,N) =F3(1)
     WRITE(6,833)(GT(J,N),SLOG(J,N),TVARG(J,N),J=2,LTAB)
  24 IF(NSEQ)2,2, 3120
3120 NSEQ = NSEQ - 1
     N1 = N1 + NADG
     GO TO 10
```

```
   30 RETURN
C COMPLETED BLOCK 8.
  100 IF(NVARG)140,140, 3130
 3130 IBLOCK = 8
      CALL REFER(NODG,NOXG,NVARG,NODE,NODES)
  140 IF(KWIT)400, 3140,400
 3140 DO 150 N = 1,NODES
       G(N) = G(N)*VOL(N)
  150 CONTINUE
  200 FORD = FOR*DELT
      IF(NVARG)300,300, 3150
 3150 CONTINUE
CCC   FIND NEW HEAT GENERATION RATES.
      DO 280 N= 1,NVARG
       J = NODG(N)
       K = NLOOK(J,6)
      IF(LTABG(N)) 3160,275,220
 3160  SET = SUMTIM + FORD
       GO TO 230
  220 SET = T(K)+FORD*DDT(K)
  230 MIN = 1
      MAX=IABS(LTABG(N))
       BET = G(J)
  240 MID = (MIN + MAX)/2
       IF(SET - TVARG(MID,N))250,270,260
  250 MAX = MID
       IF(MAX - 2)270,240,240
  260 MIN = MID
       IF(MAX - MIN - 2)270,240,240
  270 G(J) = VOL(J)*(GT(MID,N) + SLOG(MID+1,N)*(SET - TVARG(MID,N)))
      BET= 100.0*DABS((BET-G(J))/(BET+1.0E-12))
      DTMAX=AMAX1(DTMAX,BET*TVARY)
      GO TO 280
  275 SET = 0.69314718/TVARG(1,N)
      SETD=SET*DELT
      SETD=AMAX1(-60.0,AMIN1(SETD,60.0))
      SETS=AMAX1(-60.0,AMIN1(SET*SUMTIM,60.0))
      IF(ABS(SETD)-1.0E-5)3170,3170,276
 3170 SET=EXP(-SETS)*(1.0-0.5*SETD)
      GO TO 277
  276 SET=EXP(-SETS)*(1.0-EXP(-SETD))/(SET*DELT)
  277 IF(SET - 1.0E-24) 3180, 3180,278
 3180 SET = 0.
  278 G(J) = VOL(J)*GT(1,N)*SET
  280 CONTINUE
CCC   FIND TEMPERATURE CHANGE FROM INTERNAL HEAT GENERATION.
  300 DO 310 N = 1,NODES
  310 DT(N) = DT(N) + DELT*G(N)/CAP(N)
```

```
  400 RETURN                                                            06180
C COMPLETED GEN. RETURN TO HEART.                                       06190
      END                                                               06200

      SUBROUTINE SURE                                                   06210
C     CARDS COLUMN       SURE VERSION 5/29/68.                          06220
C     LIST 8                                                            06230
C     FORTRAN            SURE     TRUMP EXTERNAL HEAT FLOW SUB.FORTRAN-400 06240
CCC   VERSION 5/29/68.                                                  06250
      REAL*8 HMELT,A                                                    06260
      REAL*8 NAME,CAPS                                                  06270
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                              06280
      REAL*8 DFS                                                        06290
      REAL*8 FB                                                         06300
      REAL*8 FORD                                                       06310
      REAL*8 HEX                                                        06320
      REAL*8 FOR,SLIM,T,GEOM                                            06330
      REAL*8 CON,DA,ZIP,TRAN                                            06340
      REAL*8 TRANS                                                      06350
      REAL*8 AREAS,HSURE                                                06360
      REAL*8 TBS                                                        06370
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1      CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2      DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3      T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NP,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1      ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1      NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2      NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1      DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2      RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3      TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1      NLOOK(355,8),NODE(355),
     2      NODMAT(355),NTYPE(355),RADIUS(355),
     3      VOL(355),W(355),
     4      DFI(950),FI(950),NOD1(950),NOD2(950),
     5      F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1      LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2      TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
```

```
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      COMMON /ASURE/  AREAS(60),HSURE(60),FB(20),TBS(20),
     1       FS(60),HSURT(12,60),LTABH(60),NOXS(60),NODSB(60),

     2       NOXSB(60),POWER(60),RSURE(60),SLOH(12,60),TVARH(12,60),
     3       LTABT(20),NODB(20),SLOT(12,20),TB(20),TEMPB(12,20),
     4       TIMEB(12,20)
      COMMON /ASURES/ DFS(60),TRANS(60),NODS(60)
  810 FORMAT(8E10.3)                                                       06690
  815 FORMAT(10X,100(1H=))                                                 06700
  820 FORMAT(10X,24HEXTERNAL CONNECTION DATA)                              06710
  825 FORMAT (6I5,2E10.3,30A1)                                             06720
  826 FORMAT(/,10X,96H  NODS NODSB INDEX LTABH       POWER       DLONG     06730
     1      DRAD         HSURE         RSURE        AREAS)                 06740
  827 FORMAT(10X,4I6,4X,1P6E12.4)                                          06750
  828 FORMAT(/,19X,5HHSURT,10X,5HSLOPE,10X,5HTVARH,/,(15X,3E15.6))         06760
  830 FORMAT (2I5,10X,6E10.3)                                              06770
  831 FORMAT(//,12X,16HNODB INDEX LTABT,4X5HTEMPB,10X5HSLOPE,10X5HTIMEB)   06780
  832 FORMAT(10X,3I6,1PE15.6,15X,E15.6)                                    06790
  833 FORMAT(28X,1P3E15.6)                                                 06800
  835 FORMAT(/,10X,96H  NODS NODSB    AREAS        HSURE        POWER      06810
     $  RSURE        TRANS        HEAT FLOW    AVG RATE)                   06820
  836 FORMAT(10X,2I6,1P7E12.4)                                             06830
  838 FORMAT(10X,18HBOUNDARY NODE DATA,//,10X,5H NODB,4X,5HTEMPB,          06840
     1   8X,9HHEAT FLOW,4X,8HAVG RATE)                                     06850
  840 FORMAT(10X,I5,1P3E13.4)                                              06860
  845 FORMAT(/,10X,12HSYSTEM TOTAL,6X,1P2E13.4)                            06870
  850 FORMAT(//,18X,65HTB = TEMPB(1) + TEMPB(2)*SINF(2.*PI*(SUMTIM + TIM   06880
     1EB(2))/TIMEB(1)))                                                    06890
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)        06900
      IF(KCYC) 3000,100,200                                                06910
 3000 IF(IBLOCK - 7) 3010,40, 3010                                         06920
 3010 IF(IBLOCK - 6)700, 3020,700                                          06930
 3020 CONTINUE                                                             06940
CARD BLOCK 6.  EXTERNAL CONNECTIONS BETWEEN SURFACE AND BOUNDARY NODES.    06950
      N = NOSCON                                                           06960
      L=0                                                                  06970
      LABEL=0                                                              06980
      IF(MOE)2,3030,2                                                      06990
 3030 NOPOWS=0                                                             07000
      NORADS=NOPOWS                                                        07010
      NVARH=NORADS                                                         07020
      NOSCON=0                                                             07030
      N=0                                                                  07040
    2 READ(5,825)N1,N2,NSEQ,NADS,NADSB,L1,                                 07050
     $  DLONG,DRAD,NX1,NX2,NX3                                             07060
```

```
      IF(N1) 3040,30, 3040                                              07070
 3040 ADD = 0.0                                                         07080
      IF(DRAD) 3050,4,4                                                 07090
 3050 ADD = -(DRAD + DRADS)                                             07100
      DRAD = DRADS                                                      07110
    4 CALL PATCH(NX1,HONE,HX,LXX)                                       07120
      CALL PATCH(NX2,RONE,RX,LXX)                                       07130
      IF(RX) 3060,5, 3060                                               07140
 3060 NORADS = NORADS + 1 + NSEQ                                        07150
    5 CALL PATCH(NX3,PONE,PX,LXX)                                       07160
      IF(PX) 3070,6, 3070                                               07170
 3070 NOPOWS = NOPOWS + 1 + NSEQ                                        07180
    6 LTAB=IABS(L1)                                                     07190
      IF(LTAB - 2) 3080,7,7                                             07200
 3080 L1=0                                                              07210
      LTAB=0                                                            07220
      GO TO 11                                                          07230
    7 NVARH = NVARH + 1 + NSEQ                                          07240
      IF(LTAB - M9)8,8, 3090                                            07250
 3090 KWIT = 12                                                         07260
    8 READ(5,810)(F1(J),F2(J),J=1,LTAB)                                 07270
      DO 10 J = 2,LTAB                                                  07280
      F3(J) =  (F1(J) - F1(J-1))/(F2(J) - F2(J-1))                      07290
   10 CONTINUE                                                          07300
      F3(1) = F3(2)                                                     07310
   11 IF(MOE) 3100,15,15                                                07320
 3100 CALL SEEK2(N,N1,N2,NOXS,NOXSB,NOSCON,K)                           07330
      IF(K)14,14, 3110                                                  07340
 3110 IF(RSURE(N)) 3120,12, 3120                                        07350
 3120 NORADS = NORADS - 1                                               07360
   12 IF(LTABH(N)) 3130,13, 3130                                        07370
 3130 NVARH = NVARH - 1                                                 07380
   13 IF(POWER(N)) 3140,14, 3140                                        07390
 3140 NOPOWS = NOPOWS - 1                                               07400
   14 ITEMS(6)=MIN0(N-1,ITEMS(6))                                       07410
      GO TO 16                                                          07420
   15 NOSCON=N+1                                                        07430
      N=N+1                                                             07440
   16 IF(N-M6)18,18, 3150                                               07450
 3150 NOSCON=M6                                                         07460
      N=M6                                                              07470
      KWIT=11                                                           07480
      WRITE(6,899)N,IBLOCK                                              07490
   18 NOXS(N)=N1                                                        07500
      NODS(N)=N1                                                        07510
      NOXSB(N)=N2                                                       07520
      NODSB(N)=N2                                                       07530
      LTABH(N) = L1                                                     07540
```

```
      HSURT(1,N)=HX                                                     07550
      HSURE(N)=HX                                                       07560
      RSURE(N) = RX                                                     07570
      POWER(N) = PX                                                     07580
      DRADS = DRAD+ADD                                                  07590
      DRAD=DRADS                                                        07600
      AREAS(N) = GEOM*DLONG*DRAD**KSYM*SCALE**2                         07610
      IF(LABEL)20,20, 3160                                              07620
 3160 LABEL = 0                                                         07630
      GO TO 22                                                          07640
   20 L = L + 1                                                         07650
      IF(MOD(L-1,57))24,3170,24                                         07660
 3170 CONTINUE                                                          07670
   22 WRITE(6,826)                                                      07680
   24 WRITE(6,827)NODS(N),NODSB(N),N,LTABH(N),POWER(N),                 07690
     $ DLONG,DRAD,HSURE(N),RSURE(N),AREAS(N)                            07700
      IF(LTAB)28,28, 3180                                               07710
 3180 DO 26 J = 1,LTAB                                                  07720
      HSURT(J,N) = F1(J)                                                07730
      TVARH(J,N) = F2(J)                                                07740
      SLOH(J,N) = F3(J)                                                 07750
   26 CONTINUE                                                          07760
      LABEL = 1                                                         07770
      WRITE(6,828) (HSURT(J,N),SLOH(J,N),TVARH(J,N),J=1,LTAB)           07780
   28 IF(NSEQ)2,2, 3190                                                 07790
 3190 NSEQ = NSEQ - 1                                                   07800
      N1 = N1 + NADS                                                    07810
      N2 = N2 + NADSB                                                   07820
      GO TO 11                                                          07830
   30 RETURN                                                            07840
C COMPLETED BLOCK 6.                                                     07850
CARD BLOCK 7. EXTERNAL (BOUNDARY NODE) TEMPS, CONSTANT OR VS TIME.      07860
   40 N = NODBS                                                         07870
      LABEL = 1                                                         07880
      IF(MOE)42, 3200,42                                                07890
 3200 NVART=0                                                           07900
      NODBS=0                                                           07910
      N=0                                                               07920
   42 READ(5,830)N1,L1,(F1(J),F2(J),J=1,3)                              07930
      IF(N1) 3210,70, 3210                                              07940
 3210 IF(MOE) 3220,46,46                                                07950
 3220 CALL SEEK1(N,N1,NODB,NODBS,K)                                     07960
      IF(K)44,44, 3230                                                  07970
 3230 IF(LTABT(N)) 3240,44, 3240                                        07980
 3240 NVART = NVART - 1                                                 07990
   44 ITEMS(7)=MINO(N-1,ITEMS(7))                                       08000
      GO TO 47                                                          08010
   46 NODBS=N+1                                                         08020
```

```
      N=N+1
   47 IF(N - M7)48,48, 3250
 3250 NODBS=M7
      N=M7
      KWIT = 11
      WRITE(6,899)N,IBLOCK
   48 NODB(N) = N1
      LTABT(N) = L1
      DO 49 J = 1,3
      TEMPB(J,N) = F1(J)
      TIMEB(J,N) = F2(J)
   49 CONTINUE
      LTAB=IABS(LTABT(N))
      LTABT(N)=LTAB
      TBS(N)=TEMPB(1,N)
      TB(N)=TBS(N)
      IF(LTAB - 100)50, 3260, 3260
 3260 LTAB = 2
      IF(TIMEB(1,N)) 3270, 3270,50
 3270 TIMEB(1,N) = 1.0E24
   50 IF(LABEL)52,52, 3280
 3280 LABEL = 0
      WRITE(6,831)
   52 WRITE(6,832)NODB(N),N,LTABT(N),TEMPB(1,N),TIMEB(1,N)
      IF(LTAB - 2) 3290,55,55
 3290 LTABT(N) = 0
      GO TO 42
   55 NVART = NVART + 1
      IF(LTAB - 4)60, 3300, 3300
 3300 LABEL = 1
      READ(5,810) (TEMPB(J,N),TIMEB(J,N),J=4,LTAB)
   60 DO  65 J=2,LTAB
      SLOT(J,N) = (TEMPB(J,N) - TEMPB(J-1,N))/(TIMEB(J,N)-TIMEB(J-1,N))
   65 CONTINUE
      SLOT(1,N) = SLOT(2,N)
      WRITE(6,833) (TEMPB(J,N),SLOT(J,N),TIMEB(J,N),J=2,LTAB)
      IF(LTAB - M9)68,68, 3310
 3310 KWIT = 12
   68 IF(LTABT(N) - 100)42, 3320, 3320
 3320 WRITE(6,850)
      GO TO 42
   70 NEWBL(6) = NEWBL(6) + 1000
      RETURN
C COMPLETED BLOCK 7.
  100 IBLOCK = 6
      CALL REFER(NODS,NOXS,NOSCON,NODE,NODES)
      CALL REFER(NODSB,NOXSB,NOSCON,NODB,NODBS)
      IF(KWIT)700,3330,700
```

```
 3330 CONTINUE                                                                    08510
CCC    INITIALIZE FLUX, CHANGE NTYPE TO 1, UNLESS ALREADY 2.                      08520
       DO 130 N=1,NOSCON                                                          08530
       DFS(N)=0.0                                                                 08540
       FS(N)=0.0                                                                  08550
       N1=NODS(N)                                                                 08560
       N2=NTYPE(N1)                                                               08570
       IF(N2 - 2) 3340,130, 3340                                                  08580
 3340 NTYPE(N1) = 1                                                               08590
       IF(N2) 130, 3350,130                                                       08600
 3350 NOSPEC = NOSPEC + 1
  130 CONTINUE                                                                    08620
       FLUXS = 0.                                                                 08630
CCC    CALC CONDUCTANCE BETWEEN SURFACE AND BOUNDARY NODES.                       08640
       DO 170 N = 1,NOSCON                                                        08650
        N1 = NODS(N)                                                              08660
       N2 = NODSB(N)                                                              08670
       HSURE(N) = HSURT(1,N)                                                      08680
       T1 = TBASE + T(N1)                                                         08690
       T2 = TB(N2) + TBASE                                                        08700
       DT12=AMAX1(1.0E-24,ABS(T1-T2))                                             08710
       RAD = SIGMA*RSUPE(N)*(T1 + T2)*(T1*T1 + T2*T2)                             08720
       TRANS(N) = AREAS(N)*(RAD + HSURE(N)*DT12**POWER(N))                        08730
       ZIP(N1) = ZIP(N1) + TRANS(N)                                               08740
       FB(N2) = 0.0                                                               08750
  170 CONTINUE                                                                    08760
  200 FORD = FOR*DELT                                                             08770
       IF(KWIT)230, 3360, 3360                                                    08780
 3360 CONTINUE                                                                    08790
CCC    FIND TOTAL HEAT FLUX ACROSS EACH EXTERNAL CONNECTION.                      08800
       DO 205 N = 1,NOSCON                                                        08810
       N2 = NODSB(N)                                                              08820
       FS(N) = FS(N) + DFS(N)                                                     08830
       FB(N2) = FB(N2)+DFS(N)                                                     08840
  205 CONTINUE                                                                    08850
       IF(NOW)230,230, 3370                                                       08860
 3370  IF(KCYC - 1)230, 3380, 3380                                                08870
 3380 WRITE(6,838)                                                                08880
       TX=AMAX1(SUMTIM-TAU,1.0E-12)                                               08890
       FLUXS = 0.0                                                                08900
       DO 210 N = 1,NODBS                                                         08910
       FX = FB(N)/TX                                                              08920
       FLUXS = FLUXS + FB(N)                                                      08930
       WRITE(6,840)NODB(N),TB(N),FB(N),FX                                         08940
  210 CONTINUE                                                                    08950
       FX = FLUXS/TX                                                              08960
       WRITE(6,845)FLUXS,FX                                                       08970
       WRITE(6,815)                                                               08980
```

```
      IF(KCYC - 1)230,220, 3390                                         08990
 3390  IF(KDATA)230,230, 3400                                           09000
 3400 CONTINUE                                                          09010
  220 WRITE(6,820)                                                      09020
      DO 225 N =1,NOSCON                                                09030
      FX = FS(N)/TX                                                     09040
      IF(MOD(N-1,57))222,3410,222                                       09050
 3410 WRITE(6,835)                                                      09060
  222 WRITE(6,836)NOXS(N),NOXSB(N),AREAS(N),                            09070
     $ HSURE(N),POWER(N),RSURE(N),TRANS(N), FS(N), FX                   09080
  225 CONTINUE                                                          09090
      WRITE(6,815)                                                      09100
CCC   FIND NEW BOUNDARY NODE TEMPERATURES.                              09110
  230 IF(NVART)300,300, 3420                                            09120
 3420 DO 280 N = 1,NODBS                                                09130
      IF(LTABT(N)) 3430,280, 3430                                       09140
 3430 BET = TB(N)                                                       09150
      SET = SUMTIM + DELT                                               09160
      IF(LTABT(N) - 100)235, 3440, 3440                                 09170
 3440 ARG=6.28318561*(SET+TIMEB(2,N))/(TIMEB(1,N))                      09180
      TB(N)=SIN(ARG)                                                    09190
      TB(N) = TEMPB(1,N) + TEMPB(2,N)*TB(N)                             09200
      GO TO 275                                                         09210
  235 MIN = 1                                                           09220
      MAX = LTABT(N)                                                    09230
  240 MID = (MIN + MAX)/2                                               09240
       IF(SET - TIMEB(MID,N))250,270,260                                09250
  250 MAX = MID                                                         09260
       IF(MAX - 2)270,240,240                                           09270
  260 MIN = MID                                                         09280
       IF(MAX - MIN - 2)270,240,240                                     09290
  270 TB(N) = TEMPB(MID,N) + SLOT(MID+1,N)*(SET - TIMEB(MID,N))         09300
  275 TBS(N) = BET + FOR*(TB(N) - BET)                                  09310
      DTMAX=AMAX1(DTMAX,ABS(BET-TB(N)))                                 09320
  280 CONTINUE                                                          09330
CCC   FIND NEW SURFACE HEAT TRANSFER COEFFICIENTS.                      09340
  300 IF(NVARH)400,400, 3450                                            09350
 3450 DO 380 N = 1,NOSCON                                               09360
       IF(LTABH(N)) 3460,380,320                                        09370
 3460 SET = SUMTIM + FORD
       GO TO 330                                                        09390
  320 N1 = NODS(N)                                                      09400
       N2 = NODSB(N)                                                    09410
       K1 = NLOOK(N1,8)                                                 09420
      SET = 0.5*(TBS(N2) + T(K1) + FORD*DDT(K1))                        09430
  330 MIN = 1                                                           09440
      MAX=IABS(LTABH(N))                                                09450
      BET = HSURE(N)                                                    09460
```

```
  340 MID = (MIN + MAX)/2                                                 09470
       IF(SET - TVARH(MID,N))350,370,360                                   09480
  350 MAX = MID                                                           09490
       IF(MAX - 2)370,340,340                                              09500
  360 MIN = MID                                                           09510
      IF(MAX - MIN -2)370,340,340                                         09520
  370 HSURE(N) = HSURT(MID,N) + SLOH(MID+1,N)*(SET - TVARH(MID,N))        09530
      SHSURE=HSURE(N)                                                     09540
      BET=100.0*ABS(BET-SHSURE)/(ABS(BET)+1.0E-12)                        09550
      DTMAX=AMAX1(DTMAX,BET*TVARY)                                        09560
  380 CONTINUE                                                            09570
CCC   FIND NEW SURFACE BOUNDARY CONDUCTANCES.                             09580
  400 IF(NVARH + NORADS + NOPOWS)425,425, 3470                            09590
 3470 DO 420 N = 1,NOSCON                                                 09600
       IF(LTABH(N))410, 3480,410                                           09610
 3480 IF(RSURE(N))410, 3490,410                                           09620
 3490  IF(POWER(N)) 3500,420, 3500                                         09630
 3500 CONTINUE                                                            09640
  410 N1 = NODS(N)                                                        09650
      N2 = NODSB(N)                                                       09660
      ZIP(N1) = ZIP(N1) - TRANS(N)                                        09670
      T1 = TBASE + T(N1) + FORD*DDT(N1)                                   09680
      T2 = TBASE + TBS(N2)                                                09690
      DT12=AMAX1(1.0E-24,ABS(T1-T2))                                      09700
      RAD = SIGMA*RSURE(N)*(T1 + T2)*(T1*T1 + T2*T2)                      09710
      TRANS(N) = AREAS(N)*(RAD + HSURE(N)*DT12**POWER(N))                 09720
      ZIP(N1) = ZIP(N1) + TRANS (N)                                       09730
  420 CONTINUE                                                            09740
CCC   FIND TEMPERATURE CHANGES IN SURFACE NODES.                          09750
  425 DO 430 N=1,NOSCON                                                   09760
      N1=NODS(N)                                                          09770
      N2=NODSB(N)                                                         09780
      HEX = DELT*TRANS(N)*(TBS(N2) - T(N1))                               09790
      DT(N1) = DT(N1) + HEX/CAP(N1)                                       09800
      DF(N1) = DF(N1) + HEX                                               09810
       DFS(N) = HEX                                                        09820
  430 CONTINUE                                                            09830
  700 RETURN                                                              09840
C COMPLETED SURE. RETURN TO HEART.                                        09850
      END                                                                 09860


      SUBROUTINE FLOW                                                     13550
      REAL*8 DELF1,DELF2                                                  13560
      REAL*8 FLEX1,FLEX2,FLEX1A,FLEX2A                                    13570
      REAL*8 HMELT,A                                                      13580
      REAL*8 NAME, CAPS                                                   13590
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                13600
```

```
      REAL*8 CON,DA,ZIP,TRAN
      REAL*8 FOR,SLIM,T,GEOM
      REAL*8 FLOWN
      REAL*8 FORD
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      COMMON /AFLOWS/ FLOWN(50),NODF1(50),NODF2(50)
      COMMON /AFLOW/  DELF1(50),DELF2(50),FLAPS(50),FLOWT(12,50),
     1       FLINT(355),FLIPS(355),FLOPS(355),FLOUT(355),
     2       LTABFL(50),NOXF1(50),NOXF2(50),
     3       SLOFL(12,50),TVARFL(12,50)
      COMMON /ACHEMS/ KA(15),KAX(15),KB(15),KBX(15),
     1       B(355),BB(355),DB(355),DDB(355)
  805 FORMAT(/10X,20HMASS FLOW CONNECTION,I5,5X,5HNODES,I5,
     $   3HAND,I5,5X,27HARE OF DIFFERENT MATERIALS.)
  810 FORMAT(8E10.3)
  815 FORMAT(6I5,10A1,2E10.3)
  820 FORMAT(//,11X,24HNODF1 NODF2 INDEX LTABFL,3X,5HFLOWT,10X,
     1   5HSLOFL,10X,6HTVARFL,9X,5HDELF1,10X,5HDELF2)
  825 FORMAT(10X,4I6,1X,1PE15.6,30X,2E15.6)
  830 FORMAT(35X,3E15.6)
```

```
  835 FORMAT(10X,100(1H=))                                                     14050
  840 FORMAT(10X,14HMASS FLOW DATA,//,10X,75HSOURCE  SINK    FLOW RATE          14060
     1  NET FLOW    AVG RATE      SOURCE WT      SINK WT)                      14070
  845 FORMAT(10X,2I6,1P5E13.5)                                                 14080
  850 FORMAT(/,10X,99H  NODE    RATE IN      RATE OUT   NET FLOW IN  NET        14090
     1FLOW OUT AVG FLOW IN  AVG FLOW OUT RESIDENCE TIME)                       14100
  855 FORMAT(10X,I6,1P7E13.5)                                                  14110
  899 FORMAT(15(1H*),17HMORE THAN ALLOWED,I5,15H ITEMS IN BLOCK,I3)            14120
      IF(KCYC) 3000,100,200                                                    14130
 3000 CONTINUE                                                                 14140
CARD BLOCK 10. MASS FLOW CONNECTIONS.                                          14150
      N = NOFLOW                                                               14160
      L = 0                                                                    14170
      IF(MOE)2, 3010,2                                                         14180
 3010 NVARFL=0                                                                 14190
      NOFLOW=0                                                                 14200
      N=0                                                                      14210
    2 READ(5,815)N1,N2,NSEQ,NADF1,NADF2,L1,NX1,D1,D2                           14220
      IF(N1) 3020,70, 3020                                                     14230
 3020 DS = D1 + D2                                                             14240
      IF(DS)5, 3030,5                                                          14250
 3030 DX1 = DX2                                                                14260
      GO TO 10                                                                 14270
    5 DX1 = D2/DS                                                              14280
      DX2 = D1/DS                                                              14290
   10 CALL PATCH(NX1,FONE,FX,LXX)                                              14300
      LTAB=IABS(L1)                                                            14310
      IF(LTAB - 2) 3040,15,15                                                  14320
 3040 L1=0                                                                     14330
      LTAB=0                                                                   14340
      GO TO 30                                                                 14350
   15 IF(LTAB - M9)20,20, 3050                                                 14360
 3050 KWIT = 12                                                                14370
   20 READ(5,810)(F1(J),F2(J),J=1,LTAB)                                        14380
      DO 25 J = 2,LTAB                                                         14390
      F3(J) = (F1(J) - F1(J-1))/(F2(J) - F2(J-1))                              14400
   25 CONTINUE                                                                 14410
      F3(1) = F3(2)                                                            14420
      NVARFL = NVARFL + 1 + NSEQ
   30 IF(MOE) 3060,35,35                                                       14440
 3060 CALL SEEK2(N,N1,N2,NOXF1,NOXF2,NOFLOW,K)                                 14450
      IF(K)34,34, 3070                                                         14460
 3070 IF(LTABFL(N)) 3080,34, 3080                                              14470
 3080 NVARFL = NVARFL - 1                                                      14480
   34 ITEMS(10)=MIN0(N-1,ITEMS(10))
      GO TO 40                                                                 14500
   35 NOFLOW=N+1                                                               14510
      N=N+1                                                                    14520
```

```
   40 IF(N - M10)45,45, 3090
 3090 NOFLOW = M10
      N=M10
      KWIT = 11
      WRITE(6,899)N,IBLOCK
   45 NOXF1(N)=N1
      NODF1(N)=N1
      NOXF2(N)=N2
      NODF2(N)=N2
      LTABFL(N) = L1
      FLOWN(N) = FX
      DELF1(N) = DX1
      DELF2(N) = DX2
      L = L + 1
      IF(MOD(L-1,57))50,3100,50
 3100 WRITE(6,820)
   50 WRITE(6,825)NODF1(N),NODF2(N),N,LTABFL(N),FLOWN(N),D1,D2
      IF(LTAB)60,60, 3110
 3110 DO 55 J = 1,LTAB
      FLOWT(J,N) = F1(J)
      TVARFL(J,N) = F2(J)
      SLOFL(J,N) = F3(J)
   55 CONTINUE
      WRITE(6,830)(FLOWT(J,N),SLOFL(J,N),TVARFL(J,N),
     $ J=1,LTAB)
   60  IF(NSEQ)2,2, 3120
 3120 NSEQ = NSEQ - 1
      N1 = N1 + NADF1
      N2 = N2 + NADF2
      GO TO 30
   70 RETURN
C COMPLETED BLOCK 10.
  100 IBLOCK = 10
      CALL REFER(NODF1,NOXF1,NOFLOW,NODE,NODES)
      CALL REFER(NODF2,NOXF2,NOFLOW,NODE,NODES)
      IF(KWIT)500, 3130,500
 3130 CONTINUE
CCC   CHECK MATERIALS, WEIGHT FACTORS OF MASS FLOW CONNECTIONS.
      DO 120 N = 1,NOFLOW
      N1 = NODF1(N)
      N2 = NODF2(N)
      IF(NTYPE(N1)-2)110, 3140,110
 3140 DELF1(N)=1.0
      DELF2(N)=0.0
      GO TO 115
  110 IF(NTYPE(N2)-2)115, 3150,115
 3150 DELF2(N)=1.0
      DELF1(N)=0.0
```

```
  115 IF(NODMAT(N1) - NODMAT(N2)) 3160,120, 3160                        15010
 3160 WRITE(6,805)N,NOXF1(N),NOXF2(N)                                   15020
  120 CONTINUE                                                          15030
CCC 2(ITIALIZE NET FLOW AND FLOW RATES FOR EACH NODE.                   15040
      DO 125 N = 1,NODES                                                15050
      FLOUT(N)=0.0                                                      15060
      FLINT(N)=0.0                                                      15070
      FLIPS(N)=0.0                                                      15080
      FLOPS(N)=0.0                                                      15090
  125 CONTINUE                                                          15100
      DO 130 N = 1,NOFLOW                                               15110
      FLAPS(N) = 0.0                                                    15120
  130 CONTINUE                                                          15130
      DO 140 N = 1,NOFLOW                                               15140
      N1 = NODF1(N)                                                     15150
      N2 = NODF2(N)                                                     15160
CCC   REORCER CONNECTION IN DIRECTION OF POSITIVE MASS FLOW.            15170
      IF(FLOWN(N)) 3170,140,135                                         15180
 3170 FLOWN(N) = -FLOWN(N)                                              15190
      D1 = DELF1(N)                                                     15200
      NODF1(N) = N2                                                     15210
      NOXF1(N) = NODE(N2)                                               15220
      DELF1(N) = DELF2(N)                                               15230
      NODF2(N) = N1                                                     15240
      NOXF2(N) = NODE(N1)                                               15250
      DELF2(N) = D1                                                     15260
CCC   FIND NET FLOW RATES IN AND OUT OF EACH NODE.                      15270
  135 FLOUT(N1) = FLOUT(N1) + FLOWN(N)                                  15280
      FLINT(N2) = FLINT(N2) + FLOWN(N)                                  15290
CCC    FIND CONTRIBUTION OF MASS FLOW TO CONDUCTANCE OF EACH NODE.      15300
      ZIP(N2) = ZIP(N2) + FLOWN(N)*CAP(N2)/HEFT(N2)                     15310
  140 CONTINUE                                                          15320
CCC   START HERE DURING CYCLING.                                        15330
  200 FORD = FOR*DELT                                                   15340
      IF(KWIT)208, 3180, 3180                                           15350
 3180 DO 205 N = 1,NOFLOW                                               15360
       N1 = NODF1(N)                                                    15370
       N2 = NODF2(N)                                                    15380
      FLOWX = DELTS*FLOWN(N)                                            15390
       FLAPS(N) = FLAPS(N) + FLOWX                                      15400
      FLOPS(N1) = FLOPS(N1) + FLOWX                                     15410
       FLIPS(N2) = FLIPS(N2) + FLOWX                                    15420
  205 CONTINUE                                                          15430
  208 IF(NOW)215,215, 3190                                              15440
 3190  IF(KCYC - 1)215,210, 3200                                        15450
 3200 IF(KDATA)215,215, 3210                                            15460
 3210  IF(NVARFL)215,215, 3220                                          15470
 3220 CONTINUE                                                          15480
```

```
  210 WRITE(6,840)
      TX=AMAX1(SUMTIM-TAU,1.0E-12)
      DO 212 N = 1,NOFLOW
      FX = FLAPS(N)/TX
      WRITE(6,845)NOXF1(N),NOXF2(N),FLOWN(N),FLAPS(N),FX,
     1   DELF1(N),DELF2(N)
  212 CONTINUE
      WRITE(6,835)
      WRITE(6,850)
      DO 214 N = 1,NODES
      IF(FLINT(N))214,214, 3230
 3230 DFLOT = HEFT(N)/FLINT(N)
      FX1 = FLIPS(N)/TX
      FX2 = FLOPS(N)/TX
      WRITE(6,855)NODE(N),FLINT(N),FLOUT(N),
     $  FLIPS(N),FLOPS(N),FX1,FX2,DFLOT
  214 CONTINUE
      WRITE(6,835)
  215 IF(NVAPFL)300,300, 3240
 3240 CONTINUE
CCC   FIND NEW MASS FLOW RATES.
      DO 280 N = 1,NOFLOW
      N1 = NODF1(N)
      N2 = NODF2(N)
       K1 = NLOOK(N1,7)
      K2 = NLOOK(N2,7)
      IF(LTABFL(N)) 3250,280,220
 3250  SET = SUMTIM + FORD
       GO TO 230
  220 SET = DELF1(N)*(T(K1)+FORD*DDT(K1))+DELF2(N)*(T(K2)+FORD*DDT(K2))
  230  BET = FLOWN(N)
      ZIP(N2) = ZIP(N2) - FLOWN(N)*CAP(N2)/HEFT(N2)
      FLINT(N2) = FLINT(N2) - FLOWN(N)
      FLOUT(N1) = FLOUT(N1) - FLOWN(N)
      MIN = 1
      MAX=IABS(LTABFL(N))
  240 MID = (MIN + MAX)/2
       IF(SET - TVARFL(MID,N))250,270,260
  250 MAX = MID
       IF(MAX - 2)270,240,240
       IF(MAX - MIN - 2)270,240,240
  260 MIN = MID
CCC   FIND NEW NET FLOW RATES, CONDUCTANCE CONTRIBUTIONS OF MASS FLOW.
  270 FLOWN(N) = FLOWT(MID,N) + SLOFL(MID+1,N)*(SET - TVARFL(MID,N))
CCC   REORDER CONNECTION IN DIRECTION OF POSITIVE MASS FLOW.
      IF(FLOWN(N)) 3260,278,275
 3260 FLOWN(N) = -FLOWN(N)
      D1 = DELF1(N)
```

```
      NODF1(N) = N2                                                     15970
      NOXF1(N) = NODE(N2)                                               15980
      DELF1(N) = DELF2(N)                                               15990
      NODF2(N) = N1                                                     16000
      NOXF2(N) = NODE(N1)                                               16010
      DELF2(N) = D1                                                     16020
  275 ZIP(N2) = ZIP(N2) + FLOWN(N)*CAP(N2)/HEFT(N2)                     16030
      FLINT(N2) = FLINT(N2) + FLOWN(N)                                  16040
      FLOUT(N1) = FLOUT(N1) + FLOWN(N)                                  16050
  278 BET=100.0*DABS((BET-FLOWN(N))/(BET+1.0E-12))                      16060
      DTMAX=AMAX1(DTMAX,BET*TVARY)                                      16070
  280 CONTINUE                                                          16080
CCC   FIND TEMPERETURE CHANGES DUE TO MASS FLOW.                        16090
  300 DO 320 N = 1,NOFLOW                                               16100
      N1 = NODF1(N)                                                     16110
      N2 = NODF2(N)                                                     16120
      K11=NODMAT(N1)                                                    16130
      K12=NODMAT(N2)                                                    16140
      FLEX1=W(N1)/HEFT(N1)-FORD*DDA(N1)*HMELT(K11)                      16150
      FLEX2=W(N2)/HEFT(N2)-FORD*DDA(N2)*HMELT(K12)                      16160
      DT(N1) = DT(N1) + FLOWN(N)*DELF2(N)*(FLEX1 - FLEX2)*DELT/CAP(N1)  16170
      DT(N2) = DT(N2) + FLOWN(N)*DELF1(N)*(FLEX1 - FLEX2)*DELT/CAP(N2)
  320 CONTINUE                                                          16190
CCC   FIND CHANGE IN CONCENTRATIONS DUE TO MASS FLOW.                   16200
      IF(NKEM)500,500,3270
 3270 DO 360 N = 1,NOFLOW                                               16220
      N1 = NODF1(N)                                                     16230
      N2 = NODF2(N)                                                     16240
      K11 = NODMAT(N1)                                                  16250
      K12 = NODMAT(N2)                                                  16260
      IF(KA(K11))3280,360,3280                                          16270
 3280 IF(KA(K12))3290,360,3290                                          16280
 3290 FLOWX = DELT*FLOWN(N)                                             16290
      FLEX1A = FLOWX*(A(N1) + FORD*DDA(N1))                             16300
      FLEX1B = FLOWX*(B(N1) + FORD*DDB(N1))                             16310
      FLEX2A = FLOWX*(A(N2) + FORD*DDA(N2))
      FLEX2B = FLOWX*(B(N2) + FORD*DDB(N2))                             16320
      DA(N1) = DA(N1) + DELF2(N)*(FLEX1A - FLEX2A)/HEFT(N1)             16330
      DA(N2) = DA(N2) + DELF1(N)*(FLEX1A - FLEX2A)/HEFT(N2)             16340
      DB(N1) = DB(N1 ) + DELF2(N)*(FLEX1B - FLEX2B)/HEFT(N1)            16350
      DB(N2) = DB(N2) + DELF1(N)*(FLEX1B - FLEX2B)/HEFT(N2)             16360
  360 CONTINUE                                                          16370
  500 RETURN                                                            16380
C COMPLETED FLOW. RETURN TO HEART.                                      16390
      END                                                               16400
```

```
      SUBROUTINE SEEK2 (N, N1, N2, NB1, NB2, MAX, K)                       16410
CCC   VERSION 5/29/68.                                                     16420
CCC   SEEK2 LOOKS FOR PAIR OF NUMBERS N1, N2 INPAIR OF ARRAYD NB1, NB2,    16430
CCC    SETS N TO SUBSCRIPT OF NB1, NB2 IF FOUND, OTHERWISE SETS N TO       16440
CCC   ARRAY SIZE PLUS 1.                                                   16450
      REAL*8 NAME                                                          16460
      COMMON NAME(20)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1      ITEMS(15),NEWBL(15)
      DIMENSION NB1(1),NB2(1),NBB1(75),NBB2(75)                            16500
  800 FORMAT(5(1H*),5HITEMS,2I6,33H NOT FOUND,  ADDED TO END OF BLOCK,     16510
     1   I4,1H.)                                                           16520
      IF(MAX)125,125, 3000                                                 16530
 3000 DO 100 I=1,MAX                                                       16540
      KOF=MAX-I+1                                                          16550
      NBB1(I)=NB1(KOF)                                                     16560
  100 NBB2(I)=NB2(KOF)                                                     16570
      DO 120 J=1,MAX                                                       16580
      IF(N1-NBB1(J)) 120,3010,120                                          16590
 3010 IF(N1-NBB2(J)) 3020,130,3020                                         16600
 3020 CONTINUE                                                             16610
  120 CONTINUE                                                             16620
  125 MAX = MAX + 1                                                        16630
      J = MAX                                                              16640
      K = 0                                                                16650
      WRITE(6,800) N1,N2,IBLOCK                                            16660
  130 J=MAX-J+1                                                            16670
      N=4                                                                  16680
      K = 1                                                                16690
      RETURN                                                               16700
      END                                                                  16710


      SUBROUTINE SEEK1 (N, N1, NB1, MAX, K)                                16720
CCC   VERSION 5/29/68.                                                     16730
CCC   SEEK1 LOOKS FOR N1 IN ARRAY NB1, SETS N TO SUBSCRIPT OF NB1 IF       16740
CCC     N1 FOUND, OTHERWISE SETS N TO ARRAY SIZE PLUS 1.                   16750
      REAL*8 NAME                                                          16760
      COMMON NAME(20)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1      ITEMS(15),NEWBL(15)
      DIMENSION NB1(1),NBB1(75)                                            16800
  800 FORMAT(5(1H*),5HITEM ,I6,6X,33H NOT FOUND,  ADDED TO END OF BLOCK,   16810
     1   I4,1H.)                                                           16820
      IF(MAX)125,125, 3000                                                 16830
 3000 DO 100 I=1,MAX                                                       16840
```

```
      KOF=MAX-I+1
  100 NBB1(I)=NB1(KOF)
      DO 120 J=1,MAX
      IF(N1-NBB1(J)) 3010,130,3010
 3010 CONTINUE
  120 CONTINUE
  125 MAX = MAX + 1
        J = MAX
      K = 0
      WRITE(6,800) N1,IBLOCK
  130 J=MAX-J+1
      N=J
      K = 1
      RETURN
      END


      SUBROUTINE CLOCK1(KSEC,CLICKA)
C     SUBROUTINE CLOCK1(KSEC,CLICKA)
      KSEC=0
      CLICKA=0.0
      RETURN
      END


      SUBROUTINE CLOCK(CLOCKA,CLOCKB)
C     SUBROUTINE CLOCK(CLOCKA,CLOCKB)
      CLOCKA=0.0
      CLOCKB=0.0
      RETURN
      END


      SUBROUTINE PLOT
      REAL*8 HMELT,A
      REAL*8 CON,DA,ZIP,TRAN
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT
      REAL*8 FOR,SLIM,T,GEOM
      REAL*8 NAME,CAPS
      INTEGER PNODE
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1        ITEMS(15),NEWBL(15)
```

```
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NCDE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXF(355),NTYPES(355),TT(355)
      COMMON /APLOT/ NODEP(12),NOXEP(12)
  824 FORMAT(I4,2F10.3,I4)                                              17450
  855 FORMAT(16I5)                                                      17460
  856 FORMAT(//10X,47HNODES FOR TEMPERATURE VS TIME CRT PLOTS.  MAX =,  17470
     1   I3,/,10X,5HINDEX,10I5,/,11X,4HNODE,10I5)                       17480
  865 FORMAT(4I5,6E10.3)                                                17490
  875 FORMAT(//,12X,94HJPIC  LOGR LOGTEM LOGTIM    FRAD1      FRAD2     17500
     $  FTEMP1      FTEMP2      FTIME1      FTIME2,/10X4I6,4X1P6E12.4)  17510
  880 FORMAT( /,10X,30HWILL MAKE TEMP VS RADIUS PLOTS)                  17520
      IF(KCYC - 1) 3000,100,100                                         17530
 3000 IF(IBLOCK -11)50, 3010,50                                         17540
 3010 NUP=0                                                             17550
      NTARP=0                                                           17560
      NGRID=0                                                           17570
      ITHERM=0                                                          17580
      INTERP=0                                                          17590
      NOSTIM=1                                                          17600
      NOSTEM=1                                                          17610
      KKK=0                                                             17620
      READ(5,865) JPIC,LOGR,LOGTEM,LOGTIM,                              17630
     $ FRAD1, FRAD2, FTEMP1, FTEMP2, FTIME1, FTIME2                     17640
      IF(JPIC) 3020,50,30                                               17650
 3020 NUP=1                                                             17660
      NTARP=1                                                           17670
      WRITE(6,880)                                                      17680
   30 LOGR=IABS(LOGR)                                                   17690
      IF(LOGR - 100) 32, 3030,3030                                      17700
```

```
 3030 LOGR=MOD(LOGR,100)
      INTERP = 1
   32 JPIC=IABS(JPIC)
      LOSR=MINO(1,LOGR)
      IF(IABS(LOGTEM)-100) 34,3040,3040
 3040 NGRID = 1
      LOGTEM=MOD(LOGTEM,100)
   34 IF(LOGTEM) 3050,35,35
 3050 NOSTEM = 2
      LOGTEM = -LOGTEM
   35 LOSTEM=MINO(1,LOGTEM)
      IF(IABS(LOGTIM)-100) 36,3060;3060
 3060 ITHERM = 1
      LOGTIM=MOD(LOGTIM,100)
   36 IF(LOGTIM) 3070,38,38
 3070 NOSTIM = 2
      LOGTIM = -LOGTIM
   38 LOSTIM=MINO(1,LOGTIM)
       IF(FRAD2) 40, 3080, 40
 3080  FRAD2 = 1.0
   40 IF(FTEMP2) 41, 3090, 41
 3090  FTEMP2 = 1.0
   41 IF(FTIME2) 42, 3100, 42
 3100  FTIME2 = 1.0
   42 WRITE(6,875) JPIC,LOGR,LOGTEM,LOGTIM,FRAD1,
     $ FRAD2,FTEMP1,FTEMP2,FTIME1,FTIME2
      READ(5,855) (NODEP(J),J=1,12)
      DO 45 J=1,M11
      IF(NODEP(J)) 3110,46, 3110
 3110 NOXEP(J) = NODEP(J)
      NUP = NUP + 1
   45 CONTINUE
   46 NUP1 = NUP - NTARP
       IF(NUP1) 50, 50, 3120
 3120 WRITE(6,856)M11,(I,I=1,10),(NODEP(J),J=1,NUP1)
   50 RETURN
C COMPLETED BLOCK 11.
  100 KKK=KKK+1
C     THE DO LOOP BELOW CHANGED TO COMMENT. SEPT 1971
C     DO 300 I=1,10
C     PNODE=NODEP(I)
C     WRITE(9,824) NODE(PNODE),T(PNODE),SUMTIM,KKK
C 300 CONTINUE
  250 RETURN
      END
```

```
      SUBROUTINE PATCH (NXX,CONE,POUT,LBX)
C     CARDS COLUMN        PATCH VERSION 5/29/68.
C     FORTRAN             PATCH   TRUMP DEC PT FINDING SUB.    FORTRAN-400
CCC   VERSION 5/29/68.
CCC   PATCH STORES CONE IN POUT IF NXX CONTAINS NO DECIMAL POINT,
CCC      OTHERWISE CONVERTS NXX TO FLOATING POINT AND STORES IN POUT.
      DIMENSION NXX(10)
      REAL*8 HMELT,A
      REAL*8 CON,DA,ZIP,TRAN
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT
      REAL*8 NAME,CAPS
      REAL*8 FOR,SLIM,T,GEOM
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NRS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAR,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
      DATA IPT/'.'/
      DO 10 J=1,10
      IF(NXX(J)-IPT) 3000,20,3000
 3000 CONTINUE
   10 CONTINUE
      POUT = CONE
      LBX = 0
```

```
      GO TO 30                                                            18590
   20 PXX=0
      CALL SRCON(NXX,1,10,PXX,POUT,NONUM)
      LBX = 1                                                             18610
   30 RETURN                                                              18620
      END                                                                 18630


      SUBROUTINE REFER(LIST,LISTX,MAX,LISTR,MAR)                          18640
CCC   VERSION  5/29/68
CCC   FINDS J FOR WHICH LISTX(N) = LISTR(J), MAKES LIST(N) = J.
      DIMENSION LIST(1),LISTR(1),LISTX(1),LISTS(950)                      18670
      REAL*8 HMELT,A                                                      18680
      REAL*8 CON,DA,ZIP,TRAN                                              18690
      REAL*8 NAME,CAPS                                                    18700
      REAL*8 DELT,DDA,DDT,CAP,DF,DT,G,HEFT                                18710
      REAL*8 FOR,SLIM,T,GEOM                                              18720
      COMMON NAME(20)
      COMMON CAPS,DELT,FOR,GEOM,HMELT(15),A(355),
     1       CAP(355),CON(355),DA(355),DDA(355),DDT(355),
     2       DF(355),DT(355),G(355),HEFT(355),SLIM(355),
     3       T(355),ZIP(355),TRAN(950)
      COMMON M1,M2,M3,M4,M5,M6,M7,M8,M9,M10,M11,M12,M13
      COMMON NPS,NR,NB,MW,M,NDATA,IBLOCK,MOE,KWIT,
     1       ITEMS(15),NEWBL(15)
      COMMON NOSPEC,NMAT,NKEM,NODES,NOCON,NOSCON,NODBS,NVARG,NIT,
     1       NOFLOW,NUP1,NTABS,NVARC,NVARK,NMELT,NREACT,NVARQ,
     2       NVARZ,NVARE,NORAD,NVARH,NORADS,NOPOWS,NVART,NVARFL
      COMMON JPIC,KCYC,KD,KDATA,KSECS,KSYM,NOGEN,NOW,NPROB,NUP,NUTS
      COMMON ALONE,BONE,CLOCKA,CLOCKB,DELTS,
     1       DSTAB,DTMAX,FONE,GONE,HONE,PONE,
     2       RONE,SCALE,SIGMA,SMALL,SUMTIM,TAU,TBASE,
     3       TMAX,TMIN,TONE,TVARY
      COMMON HMELTX(15),LTABK(15),
     1       NLOOK(355,8),NODE(355),
     2       NODMAT(355),NTYPE(355),RADIUS(355),
     3       VOL(355),W(355),
     4       DFI(950),FI(950),NOD1(950),NOD2(950),
     5       F1(12),F2(12),F3(12),NX1(10),NX2(10),NX3(10),NX4(10)
      COMMON AMAT(15),CAPT(12,15),CONT(12,15),DENS(15),
     1       LTABC(15),MAT(15),SLOC(12,15),SLOK(12,15),
     2       TMELT(15),TVARC(12,15),TVARK(12,15),WT(12,15),
     3 NOXMAT(355),
     4       NODP1(75),NODP2(75),NOXP1(75),NOXP2(75),NPROP(75)
      COMMON AA(355),F(355),GG(355),H(355),NOTE(355),
     1       NOXE(355),NTYPES(355),TT(355)
  805 FORMAT(/,15(1H*),8HERROR IN,I5,17H'TH ITEM IN BLOCK,I3,             18980
     1   52H.  REFERS TO UNSPECIFIED MATL, REACTANT, OR NODE NO.,I6,1H.)  18990
```

```
      IF(MAR*MAX)150,150, 3000
 3000 IF(NEWBL(IBLOCK))150,150, 3010
 3010 N1 = 1
      IF(IBLOCK-1)150,90,3020
 3020 IF(NEWBL(IBLOCK) - 1000) 3030,3030,90
 3030 N1 = ITEMS(IBLOCK) + 1
   90 DO 95 I=1,MAR
      KOF=MAR-I+1
   95 LISTS(I)=LISTR(KOF)
  100 DO 120 N = N1,MAX
      DO 110 J=1,MAR
      IF(LISTX(N)-LISTS(J)) 3040,113,3040
 3040 CONTINUE
  110 CONTINUE
       IF(IBLOCK - 2)114, 3050,114
 3050  IF(LISTX(N)) 3060,120, 3060
 3060 CONTINUE
  114 WRITE(6,805) N,IBLOCK,LISTX(N)
      KWIT = 5
      GO TO 150
  113 J=MAR-J+1
  115 LIST(N) = J
  120 CONTINUE
  150 RETURN
CCC   END OF REFER. RETURN TO CALLER.
      END


      SUBROUTINE  SRCON(NXX,I,J,POWER,POUT,NER)
      REAL*8 POUT8
      INTEGER DIGIT(10)/'0','1','2','3','4','5','6','7','8','9'/
      INTEGER          I,J,POWER
      DIMENSION NXX(10)
      INTEGER  INT,FRAC,K
      INTEGER POINT/'.'/
      INTEGER BLNK/'    '/
      INTEGER NEX(4)/'D','E','+','-'/
      FRAC=0
      INT = 0
      NPS= 1
      II = I
      DO  1 N =II,J
      IF(NXX(N).EQ.POINT) GO TO 5
      IF(NXX(N).EQ.BLNK)GO TO 1
      IF(NXX(N).NE.NEX(4))GO TO 10
      NPS= -1
      GO TO 1
   10 CONTINUE
```

```
      DO  2 M = 1,10
      IF(NXX(N) .EQ. DIGIT(M)) GO TO 3
    2 CONTINUE
      NERR=1
      GO TO 50
    3 INT = INT*10+M-1
    1 CONTINUE
      GO TO 20
    5 II= N+1
      IF(II.GT. J) GO TO 20
      K = 0
      DO 6  N = II,J
      IF(NXX(N).EQ.BLNK)GO TO 120
      DO 7 M=1,10
      IF(NXX(N) .EQ. DIGIT(M)) GO TO 8
    7 CONTINUE
      DO 9  JF= 1,4
      IF(NXX(N).EQ.NEX(JF))GO TO 120
    9 CONTINUE
      NERR=2
      GO TO 50
    8 FRAC = FRAC*10+M-1
      K=K+1
    6 CONTINUE
   20 POUT8= FRAC
      POUT8= POUT8*10.**(-K)
      POUT=(POUT8+INT)*10.**POWER*NPS
      RETURN
  120 CONTINUE
      NSIGN= 1
      DO 130 JA= N,J
      DO 125  JG= 1,10
      IF(NXX(JA).EQ.DIGIT(JG))GO TO 150
  125 CONTINUE
      DO 130 JB= 1,4
      IF(NXX(JA).EQ.NEX(JB))GO TO 150
  130 CONTINUE
      GO TO 20
  150 CONTINUE
      DO 170 JC= JA,J
      IF(NXX(JC).EQ.NEX(4))NSIGN=-1
      IF(NXX(JC).EQ.BLNK)GO TO 170
      DO 170 JD= 1,10
      IF(NXX(JC).EQ.DIGIT(JD))GO TO 200
  170 CONTINUE
      NERR=3
      GO TO 50
  200 POWER=(JD-1)*NSIGN
```

```
      IF(JC.EQ.10)GO TO 20
      DO 210 JE= 1,10
      IF(NXX(JC+1).EQ.DIGIT(JE))POWER=POWER*10+(JE-1)*NSIGN
  210 CONTINUE
      GO TO 20
   50 WRITE(6,51) NERR,(NXX(JA),JA=I,J)
   51 FORMAT(////10X,'NERR = ',I2,//10X,'NXX = ',10A4/)
      RETURN
      END
```

BIBLIOGRAPHY

1. Lawrence Livermore Laboratory, Report UCRL-14754, Rev. II, TRUMP: A Computer Program for Transient and Steady State, Temperature Distributions in Multidimensional Systems, by Arthur L. Edwards, 1 July 1969.

2. Holman, J. P., Heat Transfer, 2nd Ed., p. 46-50, 79-81, McGraw Hill, 1968.

3. Carslaw, H. S. and Jaeger, J. C., Conduction of Heat in Solids, 2nd Ed., p. 282-286, Clarendon Press, 1959.

4. Lawrence Livermore Laboratory, Report UCRL-50816, FED: A Computer Program to Generate Geometric Input for the Heat-Transfer Code Trump, by Dale A. Schauer, 10 February 1970.

5. Jaeger, J. C., "Numerical Values for the Temperature in Radial Heat Flow", J. Math Phys, Vol. 34, p. 316-321, 1956.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Mechanical Engineering Department Library, Code 59 Naval Postgraduate School Monterey, California 93940	1
4.	Assistant Professor T. E. Cooper, Code 59Cg Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	1
5.	Thesis Office, Attn: M. Burton Naval Postgraduate School Monterey, California 93940	1
6.	Dz. Kv. Komt. Teknik Daire Baskanligi Ankara, Turkey	1
7.	Istanbul Teknik Universitesi Makina Fakultesi, Dekanligi Gumussuyu, Istanbul, Turkey	1
8.	LT(jg) Coskun Erbayram 1696 Sok. N$^{\underline{o}}$63 Karsiyaka, Izmir, Turkey	1
9.	Dr. Richard Winfrey Code L-64 Naval Civil Engineering Laboratory Port Hueneme, California 93043	1

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Naval Postgraduate School Monterey, California 93940	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

A COMPUTER PROGRAM FOR SOLVING TRANSIENT HEAT CONDUCTION PROBLEMS

4. DESCRIPTIVE NOTES *(Type of report and, inclusive dates)*

Master's Thesis, December 1971

5. AUTHOR(S) *(First name, middle initial, last name)*

Coskun Erbayram

6. REPORT DATE December 1971	7a. TOTAL NO. OF PAGES 132	7b. NO. OF REFS 5
8a. CONTRACT OR GRANT NO. b. PROJECT NO c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Naval Postgraduate School Monterey, California 93940

13. ABSTRACT

A computer program, with the code name TRUMP, was developed in 1965 by the Lawrence Livermore Laboratory to solve a variety of transient and steady-state conduction heat flow problems in simple or complicated geometric configurations. The purpose of this investigation was to adapt "TRUMP" to the Naval Postgraduate School IBM/360 Model 67 computer system. Several heat conduction problems are solved with the adapted version of TRUMP and the results compare closely with the analytical solutions to these problems. The example problem inputs may be used as guides in preparing input data for future problems. Some suggestions are also given for further development of the program.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
heat transfer						
conduction						
computer program						
numerical solution						
Trump						

Thesis 133471
E5656 Erbayram
c.1 A computer program for solving transient heat conducting problems.

3 JAN [illegible] 19971
3 APR 73 20583
1 DEC 87 33381

Thesis 133471
E5656 Erbayram
c.1 A computer program for solving transient heat conducting problems.

thesF5656
A computer program for solving transient
3 2768 002 06197 0
DUDLEY KNOX LIBRARY